

DOS III RELOCATING LOADER
(REV A)

BINARY TAPE	24308-60001
SOURCE TAPES	24308-80001
	24308-80002
	24308-80003
	24308-80004
	24308-80005
	24308-80006
	24308-80007
SOURCE LISTING	24308-90001

PAGE 0001

0001

ASMB,L,X,C,N,R,B

** NO ERRORS*

```

0001          ASMB,L,X,C,N,R,B
0003 00000    NAM LOADR,3
0004          EXT EXEC
0005          ENT LOADR
0006          SUP

```

0007*

0008*

0009* THE DISC OPERATING SYSTEM RELOCATING LOADER IS USED TO
 0010* RELOCATE BINARY PROGRAMS TO ABSOLUTE CODE THAT ARE TO BE RUN IN
 0011* THE USER AREA OF THE SYSTEM. THE RELOCATABLE BINARY MAY COME
 0012* FROM THE D.O.S. ASSEMBLER, FORTRAN, AND/OR ALGOL. THESE BINARY
 0013* PROGRAMS MAY BE LOADED FROM BINARY TAPES, JOB BINARY FILE OR USER
 0014* SPECIFIED FILE(S).

0015*

0016* PROGRAMS CAN BE MADE TO BE MADE UP OF A -MAIN- PROGRAM WITH OR
 0017* WITHOUT PROGRAM SEGMENTS. PROGRAMS CAN BE MADE TO RUN UNDER
 0018* CONTROL OF -DEBUG- ROUTINE. REQUESTED FORTRAN RELOCATABLE BINARY
 0019* LIBRARY ROUTINES ARE AUTOMATICALLY APPENDED TO THE END OF THE
 0020* PROGRAMS. A FRESH COPY OF -DEBUG- SHALL BE APPENDED TO EACH
 0021* SEGMENT BY THE LOADER IF THE DEBUG OPTION IS SELECTED. TO INSURE
 0022* THAT THE SEGMENT, AS WELL AS -MAIN- , RUNS UNDER CONTROL OF
 0023* -DEBUG-

0024*

0025* THE ABSOLUTE CODE IS PLACED ON THE DISC IMMEDIATELY FOLLOWING
 0026* THE END OF THE LAST USER FILE. THE ABSOLUTE CODE CANNOT EXCEED
 0027* THE USER AREA CORE SIZE. THE BASE PAGE LINKAGES, IF ANY ARE
 0028* REQUIRED, ARE PLACED ON THOSE SECTORS OF DISC TRACK WHICH FOLLOW
 0029* THE -MAIN- OR -SEGMENT- FOR THE PROGRAM.

0030*

0031* THE USER DIRECTORY IS UPDATED FOR EACH PROGRAM -MAIN- OR
 0032* -SEGMENT- AND THE USER AREA BOUNDARY VALUE IS UPDATED.

0034* THE RELOCATING LOADER PERMITS SEVERAL LOADING OPTIONS ENTERED
0035* VIA,
0036* :PROG,LOADR(,L,M,N,O,P)
0037*
0038* WHERE
0039*
0040* L IS SECONDARY INPUT DEVICE LOGICAL UNIT. IF L
0041* NOT SPECIFIED THEN INPUT FROM JBIN AREA ONLY.
0042* IF 2, THEN FROM JBIN AND USER FILES ONLY
0043* IF NOT 0 OR 2, THEN FROM JBIN, USER FILES, AND
0044* PAPER TAPE FROM LOGICAL UNIT L
0045* M IS LIST OUTPUT DEVICE LOGICAL UNIT NUMBER. IF NOT
0046* SPECIFIED, THEN 6 IS ASSUMED.
0047* N IS NO DEBUG/DEBUG OPTION, 0 IS ASSUMMED IF NOT
0048* SPECIFIED.
0049* O IS CURRENT/BASE PAGE
0050* LINKING OPTION, 0 IS
0051* ASSUMED IF NOT SPECIFIED.
0052* P IS PROGRAM BOUNDS OPTION, 0 IS
0053* ASSUMED IF NOT SPECIFIED.

THE RELOCATABLE LOADER FUNCTIONS AS FOLLOWS:

- (1). PROCESSING OF CONTROL INPUT SPECIFIED FILES (INCLUDES JOB BINARY FILE AND SPECIFIED USER FILES)
- (2). PROCESSING BINARY PROGRAMS INPUT FROM THE SECONDARY INPUT DEVICE.
- (3). SEARCH RELOCATABLE LIBRARY TO ATTEMPT TO SATISFY UNDEFINED EXTERNAL REFERENCES.
- (4). GENERATE ABSOLUTE CODE ON DISC FOR MAIN AND SEGMENTS AND UPDATE USER DIRECTORY AND USER-WORK AREA BOUNDARY.

TWO TABLES ARE GENERATED BY THE RELOCATING LOADER WHEN ENCOUNTERING NAM AND ENT/EXT RECORDS OF THE RELOCATABLE BINARY PROGRAMS ON USER FILE, JOB BINARY FILES, AND PROGRAMS INPUT VIA SECONDARY INPUT DEVICE. THE TABLES ARE :

- (1). IDENT TABLE FOR PROGRAM NAME NAME ENTRIES
- (2). LOADER SYMBOL TABLE FOR ENT/EXT ENTRIES

IDENT TABLE FORMAT(ID)

- WORD 1: ID1 = NAME 1,2
- WORD 2: ID2 = NAME 3,4
- WORD 3: ID3 = NAME 5, USEAGE FLAG
- WORD 4: ID4 = BIT15 IS DISC LOC FLAG; REM FOR COMMON LENGTH
- WORD 5: ID5 = DISC ADDRESS
- WORD 6: ID6 = (15)=MAIN/SUBROUTINE ; (6=0)=TYPE
- WORD 7: ID7 = DISC LENGTH ; USER MAIN IDENT ADDR FOR SEGMENTS

LOADER SYMBOL TABLE FORMAT(LST)

- WORD 1: LST1 = NAME 1,2
- WORD 2: LST2 = NAME 3,4
- WORD 3: LST3 = NAME 5, ORDINAL
- WORD 4: LST4 = IDENT ADDRESS
- WORD 5: LST5 = BP LINK ADDRESS

PROGRAM TYPES THAT MAY BE LOADED:

- 0: TREATED AS USER MAIN RELOCATABLE BINARY PROGRAMS
- 3: USER MAIN RELOCATABLE BINARY PROGRAM
- 4: DEVICE DRIVER RELOCATABLE BINARY PROGRAMS
- 5: USER SEGMENT RELOCATABLE BINARY PROGRAMS
- 6: RELOCATABLE FORTRAN LIBRARY PROGRAMS
- 7: RELOCATABLE USER BINARY UTILITY PROGRAMS

UPON COMPLETION OF TABLE GENERATION FROM STEP (1) AND (2) ABOVE, A CHECK IS MADE FOR ANY UNDEFINED EXTERNAL REFERENCES. IF THERE ARE ANY, THESE MAY BE ENTERED FROM SECONDARY INPUT DEVICE OR USER FILE, OR USER MAY SPECIFY THAT LOADING IS TO TAKE PLACE WITH UNDEFINED EXTERNALS.

A SEARCH IS MADE IN USER DIRECTORY FOR MAIN PROGRAM NAME AND SEGMENT NAMES TO BE LOADED. IF DUPLICATE NAMES OCCUR, THE LOADER IS ABORTED. THE ABSOLUTE CODE FOR MAIN IS GENERATED AND PLACED ON DISC. THE USER DIRECTORY IS UPDATED WITH THE NEW FILE ENTRY AND THE USER/WORK AREA BOUNDARY VALUE IS UPDATED. THE ABSOLUTE CODE FOR REQUESTED SEGMENTS ARE GENERATED AND FILE NAME

0111* IS ENTERED INTO DIRECTORY.
0112* ERROR DIAGNOSTICS
0113* L01 - CHECKSUM ERROR
0114* L02 - ILLEGAL RECORD
0115* L03 - MEMORY OVERFLOW
0116* L04 - BASE PAGE LINKAGE OVERFLOW
0117* L05 - SYMBOL TABLE OVERFLOW
0118* L06 - MAIN OR SEGMENT NAME ALREADY IN USER OR SYS DIRECTORY
0119* L07 - DUPLICATE ENTRY POINTS
0120* L08 - NO TRANSFER ADDRESS PRESENT
0121* L09 - RECORD OUT OF SEQUENCE
0122* L10 - OUT OF DIRECTORY OR WORK DISC SPACE
0123* L11 - PROGRAM IDENT OVERFLOW
0124* L12 - SPECIFIED FILE NOT IN USER DIRECTORY
0125* L13 - PROGRAM NAME DUPLICATION
0126* L14 - BASE PAGE LENGTH NON-ZERO
0127* L15 - SEGMENT LOADED BEFORE MAIN
0128* L16 - PROGRAM OVERLAY ERROR
0129* L17 - ILLEGAL LIBRARY RECORD
0130* L18 - ILLEGAL BASE PAGE BOUND
0131* L19 - ILLEGAL MAIN PROGRAM BOUND

0133*

0134* PROCESS PROG PARAMETERS

0135*

ADDRESS	HEX	ASCII	OPERATION	REMARKS
0136	00000	160001	LOADR LDA B,I	INPUT UNIT
0137	00001	002002	SZA	
0138	00002	033266R	IOR M300	SET FOR VARIABLE BINARY INP
0139	00003	073112R	STA PGMIN	
0140	00004	006004	INB	
0141	00005	160001	LDA B,I	LIST OUTPUT UNIT
0142	00006	002003	SZA,RSS	
0143	00007	026012R	JMP ++3	STANDARD UNIT 6
0144	00010	033176R	IOR M600	SET WRITE BITS
0145	00011	073157R	STA LISTU	
0146	00012	006004	INB	
0147	00013	160001	LDA B,I	NO DEBUG/DEBUG OPTION
0148	00014	073041R	STA DBFLG	
0149	00015	006004	INB	
0150	00016	160001	LDA B,I	CURRENT/BASE PAGE LINK OPTION
0151	00017	073155R	STA CPL	
0152	00020	006004	INB	
0153	00021	160001	LDA B,I	BOUNDS/NO BOUNDS SPECIFICATION
0154	00022	050054	CPA .1	
0155	00023	026413R	JMP STBND	YES - SET PROGRAM BOUNDS
0156	00024	060254	LDA SMFWA	NO - STORE SYSTEM BOUNDS
0157	00025	073154R	STA UMFWA	AS DEFAULT
0158	00026	060100	LDA SMLWA	
0159	00027	073155R	STA UMLWA	
0160	00030	060255	LDA SBFWA	
0161	00031	073152R	STA UBFWA	
0162	00032	060256	LDA SBLWA	
0163	00033	073153R	STA UBLWA	

0164*

0165*

0166* SUPPRESS SYSTEM SEARCH OPTION

0167*

```

0168 00034 060331 LOAD1 LDA SSFLG      CHECK SYSTEM SEARCH FLAG
0169 00035 053203R      CPA ASCNO
0170 00036 026043R      JMP LO01      = NO, SO OK
0171 00037 073127R      STA SAVSS     SAVE SS FLAG
0172 00040 002400      CLA             CLEAR SS FLAG IN BASE PAGE
0173 00041 067204R      LDB XSSFL
0174 00042 017543R      JSB BPSTR

```

0175*

0176* SET UP LST, IDENT, & BASE PAGE TABLES

0177*

```

0178 00043 060147 L001 LDA EXPG7 GET END BOUND OF LOADER
0179 00044 073205R STA BPBUF
0180*
0181 00045 063152R LDA UBFWA FWA OF USER BASE PAGE
0182 00046 073135R STA TBREL
0183 00047 073107R STA PBREL
0184 00050 003004 CMA, INA LWA OF USER BASE PAGE BUFFER =
0185 00051 073026R STA BPOIT
0186 00052 043153R ADA UBLWA # OF USER BASE PAGE AVAILABLE +
0187 00053 073020R STA BP# # OF BASE PAGE LINKAGES
0188 00054 043205R ADA BPBUF START OF BASE PAGE LINK BUFFER=

```

0189	00055	073017R	STA BPEND	END OF USER BASE PAGE BUFFER
0190	00056	002004	INA	SET LOADER SYMBOL TABLE ADDRESS
0191	00057	072772R	STA BLST	
0192	00060	072774R	STA PLST	
0193	00061	063205R	LDA BPBUF	
0194	00062	003004	CMA, INA	
0195	00063	043152R	ADA UBFWA	
0196	00064	073103R	STA NBPBF	FORM NEG BASE PAGE BUFFER ADDRESS
0197	00065	060045	LDA N6	SET PROGRAM IDENT TABLE ADDRESS
0198	00066	040100	ADA SMLWA	
0199	00067	073002R	STA BIDNT	
0200	00070	073004R	STA PIDNT	
0201	00071	063020R	LDA BP#	CLEAR BASE PAGE LINKAGE BUFFER
0202	00072	003004	CMA, INA	
0203	00073	073150R	STA WDCNT	SET CLEAR COUNTER
0204	00074	067205R	LDB BPBUF	SET CLEAR BUFFER ADDRESS
0205	00075	002400	CLA	
0206	00076	170001	STA B, I	CLEAR LOCATION
0207	00077	006004	INB	INCREMENT BUFFER ADDRESS
0208	00100	037150R	ISZ WDCNT	SKIP-DONE CLEARING
0209	00101	026076R	JMP *-3	GO CLEAR NEXT LOCATION
0210*				
0211*	VERIFY THAT THERE IS SUFFICIENT DISC AREA &			
0212*	SET INITIAL DISC POINTERS			
0213*				
0214	00102	016001X	JSB EXEC	REQUEST WORK TRACK LIMITS
0215	00103	000110R	DEF *-5	
0216	00104	000066	DEF .17	
0217	00105	011050R	DEF DSKAD	FIRST WORK TRACK ADDRESS
0218	00106	005203R	DEF TR	LAST WORK TRACK ADDRESS
0219	00107	011140R	DEF TEMP	# SECTORS/TRACK ADDRESS
0220	00110	063060R	LDA DSKAD	CHECK IF 3 WORK TRACKS AVAILABLE
0221	00111	003004	CMA, INA	
0222	00112	043203R	ADA TR	
0223	00113	070001	STA B	
0224	00114	040051	ADA N2	
0225	00115	002020	SSA	
0226	00116	026320R	JMP DRCHA	ERROR-NOT 3 WORK TRACKS
0227	00117	063203R	LDA TR	
0228	00120	002004	INA	
0229	00121	001727	ALF, ALF	
0230	00122	072767R	STA DTRTM	FORM LAST LEGAL WORK TRACK+1
0231	00123	063050R	LDA DSKAD	
0232	00124	003004	CMA, INA	
0233	00125	043203R	ADA TR	# OF SCRATCH TRACKS AVAILABLE
0234	00126	001100	ARS	DIVIDE AREA IN HALF
0235	00127	043050R	ADA DSKAD	
0236	00130	001727	ALF, ALF	TRACK ADDRESS
0237	00131	073056R	STA DSKSS	SET BIN SCRATCH
0238	00132	001727	ALF, ALF	RIGHT JUSTIFY
0239	00133	064155	LDB SYSSC	COMPARE SYS SC TO USER SC
0240	00134	054161	CPB CUDSC	
0241	00135	026141R	JMP *-4	COMPARE SO USER NOT ON SYS
0242	00136	060154	LDA DISCO	GET LAST TRACK AVAILABLE
0243	00137	010076	AND M3777	
0244	00140	002004	INA	

0245	00141 003004	CMA,INA	MAKE NEG
0246	00142 073057R	STA DSKTM	
0247	00143 060157	LDA UDNTS	USER DISC NEXT AVAILABLE T/S
0248	00144 073047R	STA DSKAB	TRACK/SECTOR ADDRESS
0249*			
0250*	INITIALIZE IDENT & LST TABLES		
0251*	PUT EXEC ENTRY IN LST		
0252*			
0253	00145 016000R	J9B INLST	INITIALIZE LIST
0254	00146 016004R	J9B LSTX	SET CURRENT LST ADDRESSES
0255	00147 000000	NOP	
0256	00150 016031R	J9B INIDX	INITIALIZE PROGRAM IDENT LIST
0257	00151 016035R	J9B IDX	SET CURRENT IDENT ADDRESSES
0258	00152 000000	NOP	
0259	00153 063233R	LDA AEXEC	PUT EXEC INTO LST
0260	00154 172775R	STA LST1,I	
0261	00155 063234R	LDA AEXEC+1	
0262	00156 172776R	STA LST2,I	
0263	00157 063235R	LDA AEXEC+2	
0264	00160 172777R	STA LST3,I	
0265	00161 002400	CLA	
0266	00162 173000R	STA LST4,I	
0267	00163 173001R	STA LST5,I	
0268	00164 062773R	LDA TLST	
0269	00165 072774R	STA PLST	

0271*

0272* PROCESS JBIN RELOCATABLE FILES

0273*

0274	00166	060101	LDA JBINS	CHECK IF ANY JOB BINARY PROGRAMS
0275	00167	002003	SZA,RSS	
0276	00170	026242R	JMP RCNTL	NO
0277	00171	073050R	STA DSKAD	
0278	00172	050102	CPA JBINC	CHECK IF ANY STORED
0279	00173	026242R	JMP RCNTL	NO
0280	00174	060102	LDA JBINC	COMPUTE # SECTORS USED IN CURRNT
0281	00175	010073	AND M177	TRACK
0282	00176	073032R	STA COUNT	
0283	00177	060102	LDA JBINC	
0284	00200	001727	ALF,ALF	
0285	00201	010074	AND M377	
0286	00202	002004	INA	
0287	00203	073203R	STA TR	
0288	00204	016001X	JBN10 JSB EXEC	CHECK IF NEXT TRACK HOLDS BINARY
0289	00205	000212R	DEF ++5	DATA
0290	00206	007173R	DEF .16	
0291	00207	000054	DEF .1	# TRACKS
0292	00210	005203R	DEF TR	STARTING TRACK
0293	00211	005202R	DEF SECT	ACTUAL TRACK
0294	00212	063202R	LDA SECT	CHECK IF STARTING = ACTUAL
0295	00213	053203R	CPA TR	
0296	00214	026221R	JMP JBN20	YES=80 BINARY TRACK
0297	00215	002003	SZA,RSS	ZERO INDICATES END OF DISC
0298	00216	026226R	JMP JBN30	NO MORE BINARY TRACKS
0299	00217	073203R	STA TR	
0300	00220	002001	RSS	
0301	00221	037203R	JBN20 ISZ TR	INCREMENT TRACK #
0302	00222	063032R	LDA COUNT	UPDATE CUMULATIVE SECTOR #
0303	00223	040116	ADA SECTR	
0304	00224	073032R	STA COUNT	
0305	00225	026204R	JMP JBN10	GO CHECK NEXT TRACK
0306	00226	002400	JBN30 CLA	READ SYSTEM DISC
0307	00227	073110R	STA PCNTT	
0308	00230	060055	LDA .2	READ SYSTEM DISC
0309	00231	073046R	STA DISCX	
0310	00232	002404	CLA,INA	YES
0311	00233	073137R	STA TDIR	SET JBIN SEARCH FLAG
0312	00234	016362R	JSB SYMBL	SET LST AND IDENT FROM JBIN AREA

0313*

0314* PROCESS USER RELOCATABLE FILES

0315*

0316	00235	002400	RCNTF CLA	SET DISC DIRECTION TO FORWARD
0317	00236	073137R	STA TDIR	
0318	00237	063112R	LDA PGMIN	CHECK IF USER FILES/PAPER TAPE
0319	00240	002003	SZA,RSS	TO BE INPUT
0320	00241	026511R	JMP RCN50	NO, SO GO LOAD
0321	00242	063067R	RCNTL LDA FIOPT	CHECK IF FILE TO BE ENTERED FROM
0322	00243	002003	SZA,RSS	SYSTEM TELETYPE
0323	00244	026251R	JMP RCN02	MAYBE=80 MAKE FURTHER CHECK
0324	00245	060112	LDA SYSTY	YES
0325	00246	033267R	IOR M400	SET ECHO BIT
0326	00247	073140R	STA TEMP	

0327	00250	026261R	JMP	RCN04	GO OUTPUT MESSAGE
0328	00251	060111	RCN02	LDA BATCH	COMPARE BATCH AND SYS TTY DEVICE
0329	00252	050112		CPA SYSTY	
0330	00253	033267R		IOR M400	SET ECHO BIT
0331	00254	073140R		STA TEMP	
0332	00255	060111		LDA BATCH	CHECK IF BATCH IS SYS TELETYPE
0333	00256	050112		CPA SYSTY	
0334	00257	002001		RSS	YES
0335	00260	026267R		JMP RCN05	NO
0336	00261	016001X	RCN04	JSB EXEC	OUTPUT MESSAGE:
0337	00262	000267R		DEF ++5	-ENTER FILE NAME(S) OR /E
0338	00263	000055		DEF .2	
0339	00264	000054		DEF .1	
0340	00265	000346R		DEF CMES	
0341	00266	007171R		DEF .12	
0342	00267	016001X	RCN05	JSB EXEC	REQUEST INPUT FROM BATCH OR SYS
0343	00270	000275R		DEF ++5	
0344	00271	000054		DEF .1	READ REQUEST
0345	00272	011140R		DEF TEMP	CONTROL WORD
0346	00273	010726R	DATB	DEF DATAB	DATA BUFFER ADDRESS
0347	00274	007166R		DEF N72	CHARACTER COUNT
0348	00275	006003		SZB, RSS	SKIP-INPUT IN
0349	00276	026242R		JMP RCNTL	GO TO READ NEXT CONTROL INPUT
0350	00277	007004		CMB, INB	
0351	00300	076365R		STB NEGPM	SAVE NEG INPUT COUNT
0352	00301	062273R		LDA DATB	
0353	00302	000066		CLE, ELA	
0354	00303	072366R		STA PMBUF	SET INPUT BUF CHAR ADDR
0355	00304	062726R		LDA DATAB	CHECK IF INPUT IS /E
0356	00305	053212R		CPA SLSHE	
0357	00306	026511R		JMP RCN50	YES = SO DONE
0358	00307	016231R	RCN10	JSB GETP	GET ONE CONTROL PARAMETER
0359	00310	073141R		STA TERMN	SAVE TERMINATOR FLAG
0360	00311	053212R		CPA SLSHE	
0361	00312	026511R		JMP RCN50	
0362	00313	016001X		JSB EXEC	CHECK IF FILE IN USER DIRECTORY
0363	00314	000320R		DEF ++4	
0364	00315	007174R		DEF .18	
0365	00316	000214R		DEF PBUF	ADDRESS OF FILE NAME
0366	00317	011032R		DEF COUNT	# OF SECTORS IN FILE
0367	00320	073050R		STA DSKAD	SAVE TRACK/SECTOR ADDR OF FILE
0368	00321	002400		CLA	
0369	00322	073137R		STA TDIR	
0370	00323	063032R		LDA COUNT	
0371	00324	002002		SZA	
0372	00325	026334R		JMP RCN30	PRINT UNDEFINED FILE ERROR CODE
0373	00326	063176R		LDA ERR12	
0374	00327	016114R		JSB ERROR	PRINT UNDEFINED FILE NAME
0375	00330	060060		LDA .5	
0376	00331	067303R		LDB DFPBF	
0377	00332	016101R		JSB SYOUT	
0378	00333	026371R		JMP ABORT	ABORT LOADER
0379	00334	016316R	RCN30	JSB PROGL	CHECK IF PROGRAMS SPECIFIED
0380	00335	017537R		JSB SETCR	READ USER DISC
0381	00336	016362R		JSB SYMBL	ENTER ENT/EXT-PROGRAM IDENTIS
0382	00337	063141R		LDA TERMN	GET TERMINATION SYMBOL

0383	00340	053212R	CPA SLSHE	END OF PARAMETERS?
0384	00341	026511R	JMP RCN50	YES
0385	00342	062365R	LDA NEGPM	
0386	00343	002003	SZA,RSS	END OF STATEMENT?
0387	00344	026242R	JMP RCN1L	YES-READ NEXT STATEMENT
0388	00345	026307R	JMP RCN10	GO TO PROCESS NEXT PARAMETER
0389	00346	042516	CMES ASC 12,ENTER	FILE NAME(S) OR /E

```

0391*
0392*      SYMBL ROUTINE READS USER FILE OR JBIN FILE AND ADDS THESE
0393*      PROGRAMS TO THE PROGRAM LIST AND ALL ENT/EXT TO THE LST.
0394*      THE OPTION EXISTS TO ENTER ALL PROGRAMS FROM THE GR JUST
0395*      THOSE SPECIFIED. ALSO IS JBIN AREA, TRACK ADDRESS IS DECREMENTED
0396*      WHEN CROSSING TRACK BOUNDARIES RATHER THAN INCREMENTING.
0397*      CALLING SEQUENCE:
0398*          PRIOR TO ENTERING SET
0399*              DSKAD = STARTING TRACK/SECTOR OF FILE
0400*              COUNT = # SECTORS IN FILE
0401*              PCNTT = 0 IF ALL PROGRAMS IN FILE TO BE LOADED
0402*                   = N IF SPECIFIC PROGRAMS IN FILE TO BE LOADED
0403*              PLIST = PROGRAM NAMES LIST OF PROGRAMS TO BE LOADED
0404*                   (PCNTT NOT ZERO)
0405*              TDIR  = 0 IF USER FILE
0406*                   = 1 IF JBIN AREA
0407*
0408 00362 000000 SYMBL NOP
0409*
0410 00363 062362R      LDA SYMBL      SET UP RETURN ADDRESS FOR BLOCK
0411 00364 016007R      JSB SETUP      ROUTINE
0412 00365 063050R      LDA DSKAD
0413 00366 073532R      STA DSKNM      SET INITIAL NAM RECORD TR/SECT
0414 00367 072074R      STA BLKAD      SET INITIAL DISC ADDRESS
0415*
0416 00370 016022R      JSB BLOCK      INPUT 1ST BLOCK OF CODE
0417*
0418 00371 006400 SYM10 CLB      CLEAR WORDS/PROGRAM
0419 00372 076510R      STB W,PRG      COUNT
0420 00373 016077R      JSB FETCH      GET RECORD TYPE
0421 00374 050054      CPA .1      NAME RECORD?
0422 00375 002001      RSS          YES
0423 00376 026637R      JMP RCERR
0424*
0425 00377 063110R      LDA PCNTT      CHECK PROGRAMS SPECIFIED FLAG
0426 00400 002003      SZA,RSS          SKIP=YES
0427 00401 026443R      JMP SYM50
0428 00402 003004      CMA,INA
0429 00403 073150R      STA WDCNT      SET # PROG IN PLIST
0430 00404 063304R      LDA PLAD
0431 00405 073016R      STA AMAD      SET SEARCH ADDRESS
0432 00406 067035R SYM20 LDB CURAL    "B" POINTING TO
0433 00407 044055      ADB .2      PROGRAM NAME
0434 00410 163016R      LDA AMAD,I    CHECK IF NAME IN PLIST TABLE
0435 00411 150001      CPA B,I
0436 00412 002001      RSS          NAME 1,2 IN TABLE
0437 00413 026431R      JMP SYM30      NOT CURRENT PROG NAME
0438 00414 037016R      ISZ AMAD
0439 00415 006004      INB
0440 00416 163016R      LDA AMAD,I
0441 00417 150001      CPA B,I
0442 00420 002001      RSS          NAME 3,4 IN TABLE
0443 00421 026432R      JMP SYM30+1    NOT CURRENT PROG NAME
0444 00422 037016R      ISZ AMAD
0445 00423 006004      INB
0446 00424 160001      LDA B,I

```

0447	00425	010075		AND M1774	
0448	00426	153016R		CPA AMAD,I	
0449	00427	026443R		JMP SYM50	PROGRAM IN PLIST TABLE
0450	00430	026433R		JMP SYM30+2	
0451	00431	037016R	SYM30	ISZ AMAD	INCR PLIST ADDRESS
0452	00432	037016R		ISZ AMAD	INCR PLIST ADDRESS
0453	00433	037016R		ISZ AMAD	INCR PLIST ADDRESS
0454	00434	037150R		ISZ WDCNT	INCR PROG COUNT
0455	00435	026406R		JMP SYM20	GO TRY NEXT PROGRAM IN PLIST
0456	00436	016205R	SYM40	JSB SKIP	SKIP OVER REST OF RECORD
0457	00437	016077R		JSB FETCH	
0458	00440	050060		CPA .5	END RECORD?
0459	00441	026477R		JMP SYM75	
0460	00442	026436R		JMP SYM40	NO=80 SKIP OVER REST OF RECORD
0461*					
0462	00443	017375R	SYM50	JSB NAM	PROCESS NAM RECORD
0463*					
0464	00444	016205R	SYM70	JSB SKIP	SKIP OVER REST OF RECORD
0465	00445	016077R		JSB FETCH	
0466	00446	050055		CPA .2	ENT RECORD?
0467	00447	026503R		JMP SYENT	PROCESS ENTRY RECORD
0468	00450	050056		CPA .3	DBL RECORD?
0469	00451	026501R		JMP SYDBL	YES = FIND MAX
0470	00452	050057		CPA .4	EXT RECORD?
0471	00453	026504R		JMP SYEXT	PROCESS EXTERNAL RECORD
0472	00454	050060		CPA .5	END RECORD?
0473	00455	002001		RSS	
0474	00456	026637R		JMP RCERR	ILLEGAL RECORD TYPE
0475	00457	017736R		JSB END	PROCESS END RECORD
0476*					
0477	00460	062510R		LDA W,PRG	GET # WORDS IN PROG
0478	00461	040057		ADA .4	ADD 4 MORE FOR REST OF END
0479	00462	006400		CLB	(FOR DIVIDE)
0480	00463	016002X		DIV .128	COMPUTE # SECTOS IN PROGRAM
0481	00465	006002		SZB	FROM # WORDS
0482	00466	002004		INA	
0483	00467	072507R		STA #SECT	
0484*					
0485	00470	163014R		LDA ID7,I	STORE #TRACKS OR MAIN SEG
0486	00471	002003		SZA,RSS	
0487	00472	062507R		LDA #SECT	GET CURRENT PROG SECTOR COUNT
0488	00473	067137R		LDB TDIR	FORWARD OR BACKWARD DIRECTION?
0489	00474	006002		SZB	FORWARD
0490	00475	003004		CMA,INA	BACKWARD=COMPLEMENT # SECTORS
0491	00476	173014R		STA ID7,I	
0492*					
0493	00477	016205R	SYM75	JSB SKIP	SKIP REST OF END RECORD
0494	00500	026371R		JMP SYM10	PROCESS NEXT PROGRAM
0495*					
0496*					
0497	00501	017533R	SYDBL	JSB DBL	PROCESS DBL RECORD
0498	00502	026444R		JMP SYM70	
0499*					
0500	00503	003401	SYENT	CCA,RSS	ENT
0501	00504	002400	SYEXT	CLA	EXT
0502	00505	017554R		JSB EN,XT	PROCESS ENT/EXT RECORDS

0503 00506 026444R JMP SYM70

0504*

0505 00507 000000 #SECT NOP

SECTORS/PROGRAM

0506 00510 000000 W.PRG NOP

WORDS/PROGRAM

```

0508*
0509*  PROCESS RELOCATBLE BINARY INPUT
0510*
0511  00511 060055 RCN50 LDA .2      SET FOR SYSTEM DISC
0512  00512 073046R      STA DISCX
0513*
0514  00513 063056R      LDA DSKSS    SET SCRATCH AREA AS DISC ADDRESS
0515  00514 073050R      STA DSKAD    FOR OUTPUTTING RELOCATABLES
0516*
0517  00515 063112R      LDA PGMIN    CHECK IF ANY INPUT FROM SECONDAR
0518  00516 002003      SZA,RSS      INPUT DEVICE
0519  00517 026760R      JMP LODLB    NO-GO CHECK LIBRARY PROGRAM
0520  00520 010072      AND M77
0521  00521 050055      CPA .2
0522  00522 026760R      JMP LODLB    NO PAPER TAPE TO BE LOADED
0523*
0524*  READ IN BINARY RELOCATABLE INPUT FROM SECONDARY DEVICE AND
0525*  STORE IN PACKED FORMAT ONTO DISC.
0526*  NAM RECORDS ARE ENTERED INTO PROG NAME NAME TABLE AND ENT/EXT
0527*  ARE ENTERED INTO LOADER SYMBOL TABLE.
0528*
0529  00523 003400 RCN60 CCA
0530  00524 073114R      STA PLFLG    SET LOADING FLAG
0531*
0532  00525 063272R      LDA N128
0533  00526 073044R      STA DCNT     INITIALIZE DBUF COUNT
0534  00527 066535R      LDB AABUF
0535  00530 077033R      STB CURAD    INITIALIZE DBUF POINTER
0536  00531 016266R      JSB BUFCL   CLEAR DBUF
0537  00532 002400      CLA          SET FOR 1ST OUTPUT BUFFER
0538  00533 072512R      STA ABFLG
0539  00534 072220R      STA FIRST
0540*
0541  00535 063004R      LDA PIDNT
0542  00536 072657R      STA SIDNT    OF CHECKSUM ERROR
0543  00537 062774R      LDA PLST     SAVE END OF LST TABLE
0544  00540 072656R      STA SLST
0545  00541 063050R      LDA DSKAD
0546  00542 072655R      STA SDSK
0547*
0548  00543 060065 TREAD LDA .10      PRINT -LOAD TAPE- ON SYS TTY
0549  00544 066660R      LDB MESLD
0550  00545 016101R      JSB SYOUT
0551  00546 016001X      JSB EXEC     SUSPEND PROGRAM
0552  00547 000551R      DEF ++2
0553  00550 000062      DEF .7
0554  00551 160001      LDA B,I
0555  00552 002002      SZA
0556  00553 026760R      JMP LODLB    GO TO CHECK IF LOAD LIBRARY
0557  00554 003400      CCA          SET EOT FLAG TO IGNORE EOT
0558  00555 073064R      STA ETFLG
0559*
0560  00556 016001X LDRIN JSB EXEC     READ INPUT FROM 2NDARY INPUT DEV
0561  00557 000564R      DEF ++5      DEVICE INTO LBUF
0562  00560 000054      DEF .1
0563  00561 011112R      DEF PGMIN

```

0564	00562	007726R	DEF LBUF	
0565	00563	007273R	DEF .128	
0566*				
0567	00564	012566R	AND M240	
0568	00565	002302	CCE, SZA	END OF FILE/TAPE?
0569	00566	000240	M240 OCT 240	-YES (CLE INSTRUCTION)
0570	00567	006042	SEZ, SZB	REC LENGTH=0?
0571	00570	026575R	JMP TESTR	YES, SO GO PROCESS RECORD
0572	00571	063064R	LDA ETFLG	NO, SO CHECK END OF TAPE
0573	00572	002002	SZA	
0574	00573	026556R	JMP LDRIN	GO READ IN NEXT RECORD
0575	00574	026543R	JMP TREAD	OUTPUT "LOAD" AND READ IN RECORD
0576*				
0577	00575	002400	TESTR CLA	SET EOT FLAG = VALID EOT
0578	00576	073064R	STA ETFLG	
0579*				
0580	00577	067276R	LDB ALBUF	
0581	00600	077035R	STB CURAL	INITIALIZE LBUF POINTER
0001	00601	063272R	LDA N128	
0002	00602	073077R	STA LCNT	INITIALIZE LBUF COUNT
0003*				
0004	00603	016077R	JSB FETCH	GET RECORD TYPE, CHK VALIDITY
0005*				
0006	00604	073147R	STA TEMP1	SAVE RECORD TYPE
0007	00605	163035R	LDA CURAL, I	COMPUTE CHECKSUM
0008	00606	067035R	LDB CURAL	
0009	00607	044055	ADB .2	
0010	00610	140001	ADA B, I	
0011	00611	006004	INB	
0012	00612	037150R	ISZ WDCNT	
0013	00613	026610R	JMP *-3	
0014*				
0015	00614	067035R	LDB CURAL	COMPUTED = ACTUAL CHECKSUM?
0016	00615	006004	INB	
0017	00616	150001	CPA B, I	
0018	00617	002001	RSS	YES =
0019	00620	026643R	JMP CHKER	= CHECKSUM ERROR
0020	00621	063147R	LDA TEMP1	RESTORE RECORD TYPE
0021*				
0022	00622	067114R	LDB PLFLG	
0023	00623	050054	CPA .1	TYPE = NAM RECORD ?
0024	00624	026666R	JMP NAMR	YES
0025	00625	006002	SZB	
0026	00626	026641R	JMP NMERR	RECORD OUT OF SEQUENCE
0027	00627	050055	CPA .2	TYPE = ENT RECORD ?
0028	00630	026703R	JMP ENTR	YES
0029	00631	050056	CPA .3	TYPE=DBL RECORD?
0030	00632	026701R	JMP DBLR	YES
0031	00633	050057	CPA .4	TYPE = EXT RECORD ?
0032	00634	026704R	JMP EXTR	YES
0033	00635	050060	CPA .5	
0034	00636	026707R	JMP ENDR	
0035*				
0036	00637	063164R	RCERR LDA ERR02	ILLEGAL RECORD ERROR
0037	00640	026370R	JMP ABRTX	
0038*				

0039	00641	063173R	NMERR	LDA	ERR09	RECORD OUT OF SEQUENCE
0040	00642	026370R		JMP	ABRTX	
0041*						
0042	00643	063163R	CHKER	LDA	ERR01	
0043	00644	016114R		JSB	ERROR	OUTPUT L01 MESSAGE
0044	00645	062657R		LDA	SIDNT	RESTORE IDENT & LST TABLES TO
0045	00646	073004R		STA	PIDNT	THEIR INITIAL STATE
0046	00647	062656R		LDA	SLST	
0047	00650	072774R		STA	PLST	
0048	00651	062655R		LDA	SDSK	
0049	00652	073050R		STA	DSKAD	
0050	00653	016375R		JSB	SET	SET SIMULATED END-OF-TAPE
0051	00654	026523R		JMP	RCN60	TRY ENTERING PROGRAM AGAIN
0052*						
0053	00655	000000	SDSK	NOP		SAVED DISC ADDRESS
0054	00656	000000	SLST	NOP		SAVED END OF LST TABLE
0055	00657	000000	SIDNT	NOP		SAVED END OF IDENT TABLE
0056	00660	000661R	MESLD	DEF	*+1	
0057	00661	046117		ASC	5,LOAD	TAPE
0059	00666	006003	NAMR	SZB,RSS		
0060	00667	026641R		JMP	NMERR	RECORD OUT OF SEQUENCE
0061*						
0062	00670	006400		CLB		
0063	00671	077063R		STB	DSCNT	CLEAR DISC SEGMENT COUNT
0064	00672	077031R		STB	CNFLG	CLEAR DISC SEGMENT COUNT FLAG
0065	00673	077114R		STB	PLFLG	SET PLFLG = NOT LOADING
0066	00674	063050R		LDA	DSKAD	SET DISC ADDRESS OF NAM RECORD
0067	00675	073532R		STA	DSKNM	FOR DIRECTORY ENTRY.
0068*						
0069	00676	017375R		JSB	NAM	PROCESS NAM RECORD
0070*						
0071	00677	016715R	OUTR	JSB	DWRIT	PACK RECORD, OUTPUT TO DISC
0072	00700	026556R		JMP	LDRIN	GO READ NEXT RECORD
0074	00701	017533R	DBLR	JSB	DBL	PROCESS DBL RECORD
0075	00702	026677R		JMP	OUTR	OUTPUT DBL RECORD TO DISC
0077	00703	003401	ENTR	CCA,RSS		
0078	00704	002400	EXTR	CLA		
0079	00705	017554R		JSB	EN,XT	PROCESS ENT/EXT RECORDS
0080	00706	026677R		JMP	OUTR	OUTPUT RECORD TO DISC
0082	00707	003400	ENDR	CCA		
0083	00710	073114R		STA	PLFLG	
0084	00711	073031R		STA	CNFLG	
0085	00712	016715R		JSB	DWRIT	WRITE END RECORD ON DISC
0086*						
0087	00713	017736R		JSB	END	PROCESS END RECORD
0088	00714	026556R		JMP	LDRIN	

```

0090*
0091*      DWRIT PACKS THE CONTENTS OF LBUF INTO DBUF AND DUMPS
0092*      DBUF WHEN IT CONTAINS 128 WORDS OF RELOCATABLE INPUT.
0093*
0094 00715 000000 DWRIT NOP
0095 00716 067276R LDB ALBUF      GET ADDRESS OF LBUF
0096 00717 077035R STB CURAL
0097 00720 063726R LDA LBUF      GET RECORD LENGTH
0098 00721 001727 ALF,ALF
0099 00722 003004 CMA,INA
0100 00723 073077R STA LCNT
0101 00724 163035R LTOD LDA CURAL,I      GET WORD FROM LBUF AND
0102 00725 173033R STA CURAD,I      SET WORD INTO DBUF
0103 00726 037044R ISZ DCNT
0104 00727 026732R JMP GETL
0105 00730 016147R JSB DDOUT      OUTPUT IF BUFFER FULL
0106 00731 002001 RSS
0107 00732 037033R GETL ISZ CURAD      INCR DBUF ADDRESS
0108 00733 037035R ISZ CURAL      INCR LBUF ADDRESS
0109 00734 037077R ISZ LCNT
0110 00735 026724R JMP LTOD      GO MOVE NEXT WORD
0111 00736 063031R LDA CNFLG      TEST END FLAG
0112 00737 002003 SZA,RSS      SKIP IF END RECORD READ
0113 00740 026753R JMP MAXWK      CHECK FOR OVERFLOW
0114 00741 063044R LDA DCNT      CHECK IF ANY DATA IN BUFFER
0115 00742 053272R CPA N128
0116 00743 002001 RSS
0117 00744 016147R JSB DDOUT      OUTPUT END RECORD TO DISC
0118 00745 163013R LDA ID6,I      CHECK PROGRAM TYPE
0119 00746 010073 AND M177
0120 00747 050060 CPA ,5
0121 00750 026753R JMP MAXWK      USER SEGMENT - SO CHECK FOR OVFL
0122 00751 063063R LDA DSCNT
0123 00752 173014R STA ID7,I      STORE SECTOR COUNT IN IDENT
0124*
0125*      CHECK IF RELOCATABLE BINARY EXCEEDS AVAILABLE WORK TRACK SPACE
0126*
0127 00753 062757R MAXWK LDA DTRTM      MAX + 1
0128 00754 053050R CPA DSKAD      CURRENT
0129 00755 026320R JMP DRCHA      ERROR-OUT OF WORK TRACKS
0130 00756 126715R JMP DWRIT,I      RETURN
0131*
0132 00757 000000 DTRTM NOP

```

```

0134*
0135*  UPDATE LST, CHECK FOR UNDEFINED EXT &
0136*      LOAD LIBRARY ROUTINES IF NECESSARY
0137*
0138  00760 063041R LODLB LDA DBFLG
0139  00761 002003      SZA,RSS          ENTER DEBUF INTO LST
0140  00762 027003R      JMP NODBG        NO DEBUG OPTION
0141  00763 016000R      JSB INLST
0142  00764 016004R      JSB LSTX
0143  00765 026767R      JMP ++2
0144  00766 026764R      JMP *-2
0145  00767 063217R      LDA CHRDE        SET -DEBUG-- NAME INTO LST AND ID
0146  00770 172775R      STA LST1,I
0147  00771 063220R      LDA CHRDE+1
0148  00772 172776R      STA LST2,I
0149  00773 063221R      LDA CHRDE+2
0150  00774 010075      AND M1774
0151  00775 172777R      STA LST3,I
0152  00776 002400      CLA              CLEAR BASE PAGE LINK ADDRESS
0153  00777 173000R      STA LST4,I
0154  01000 173001R      STA LST5,I
0155  01001 062773R      LDA TLST
0156  01002 072774R      STA PLST
0157  01003 002400      NODBG CLA
0158  01004 073065R      STA EXCNT
0159  01005 016000R      LOADL JSB INLST
0160  01006 016004R      JSB LSTX
0161  01007 027054R      JMP LODLL
0162  01010 063233R      LDA AEXEC
0163  01011 152775R      CPA LST1,I
0164  01012 002001      RSS
0165  01013 027036R      JMP LODLX
0166  01014 063234R      LDA AEXEC+1
0167  01015 152776R      CPA LST2,I
0168  01016 002001      RSS
0169  01017 027036R      JMP LODLX
0170  01020 162777R      LDA LST3,I
0171  01021 010075      AND M1774
0172  01022 053235R      CPA AEXEC+2
0173  01023 002001      RSS
0174  01024 027036R      JMP LODLX
0175  01025 002400      CLA
0176  01026 173000R      STA LST4,I
0177  01027 063102R      LDA GOCHK        GET LINK FOR EXEC EXT
0178  01030 013201R      AND 01777
0179  01031 173001R      STA LST5,I
0180  01032 162777R      LDA LST3,I
0181  01033 010075      AND M1774
0182  01034 172777R      STA LST3,I
0183  01035 027006R      JMP LOADL+1
0184  01036 002400      LODLX CLA
0185  01037 173001R      STA LST5,I
0186  01040 162777R      LDA LST3,I
0187  01041 010075      AND M1774
0188  01042 172777R      STA LST3,I
0189  01043 163000R      LDA LST4,I

```

0190	01044	002020	SSA	
0191	01045	027050R	JMP ++3	
0192	01046	002002	SZA	
0193	01047	027006R	JMP LOADL+1	
0194	01050	037065R	ISZ EXCNT	
0195	01051	003400	CCA	
0196	01052	173000R	STA LST4,I	
0197	01053	027006R	JMP LOADL+1	
0198	01054	063065R	LODLL LDA EXCNT	
0199	01055	002003	SZA,RSS	
0200	01056	026216R	JMP LMAIN	
0201	01057	016000R	JSB INLST	INIT. LST ENTRY
0202	01060	016004R	JSB LSTX	GET NEXT LST
0203	01061	026216R	JMP LMAIN	NO UNDEFINES: GO LOAD
0204	01062	163000R	LDA LST4,I	CHECK IF UNDEFINED EXT
0205	01063	050052	CPA N1	
0206	01064	002001	RSS	YES: SEARCH LIBRARY
0207	01065	027060R	JMP --5	TRY NEXT LST
0208	01066	002400	CLA	
0209	01067	073065R	STA EXCNT	
0210	01070	073063R	STA DSCNT	
0211	01071	060055	LDA .2	SET FOR SYSTEM DISC
0212	01072	073046R	STA DISCX	
0213	01073	017126R	JSB LIBS	LOAD REQUESTED LIBRARY ROUTINES
0214	01074	063065R	LDA EXCNT	GET # LIB ROUTINES LOADED
0215	01075	002002	SZA	NONE, SO DONE LOADING ROUTINES
0216	01076	027054R	JMP LODLL	
0217	01077	016372R	JSB LSTEX	LIST UNDEFINED EXTERNALS
0218	01100	002001	RSS	UNDEFINED EXT FOUND
0219	01101	026216R	JMP LMAIN	NO UNDEFINED EXT FOUND
0220	01102	016001X	GOCHK JSB EXEC	SUSPEND LOADER
0221	01103	001105R	DEF ++2	
0222	01104	000062	DEF .7	
0223	01105	160001	LDA B,I	CHECK IF :GO,P,N
0224	01106	002002	SZA	
0225	01107	027117R	JMP CFILE	
0226	01110	006004	INB	
0227	01111	160001	LDA B,I	CHECK IF N SPECIFIED
0228	01112	002003	SZA,RSS	
0229	01113	026523R	JMP RCN60	GO INPUT FROM SECONDARY DEVICE
0230	01114	033266R	IOR M300	
0231	01115	073112R	STA PGMIN	SET INPUT LOGICAL UNIT
0232	01116	026523R	JMP RCN60	GO INPUT FROM SECONDARY DEVICE
0233	01117	050055	CFILE CPA .2	CHECK IF MORE FILES TO BE INPUT
0234	01120	027124R	JMP CF10	YES
0235	01121	050054	CPA .1	CHECK IF TO IGNORE UNDEFINES
0236	01122	026216R	JMP LMAIN	YES
0237	01123	027102R	JMP GOCHK	GET :GO INPUT
0238	01124	037067R	CF10 ISZ FIOPT	
0239	01125	026242R	JMP RCNTL	
0240*				
0241*				
0242*				
0243*				
0244	01126	000000	LIBS NOP	
0245	01127	063126R	LDA LIBS	SET RETURN ADDRESS FOR BLOCK

0246	01130	016007R		JSB SETUP	ROUTINE
0247*					
0248	01131	003400		CCA	
0249	01132	073073R		STA IGNOR	
0250	01133	060325		LDA DSCL#	SET # OF
0251	01134	003004		CMA, INA	LIBRARY ROUTINES
0252	01135	073333R		STA #LIB	
0253*					
0254	01136	063203R		LDA M0760	SET MAX # LIBRARY SECTORS
0255	01137	073032R		STA COUNT	TO 76000B
0256	01140	060324		LDA DSCLB	SET INITIAL LIBRARY TRACK/SECTOR
0257	01141	072074R		STA BLKAD	DISC ADDRESS
0258	01142	073532R		STA DSKNM	SET INITIAL NAM RECORD TR/SECT
0259*					
0260	01143	016022R	LIRD	JSB BLOCK	INPUT 1ST BLOCK OF RELOCATABLES
0261*					
0262	01144	016077R	LIB1	JSB FETCH	GET RECORD TYPE
0263	01145	050054		CPA .1	CHECK IF NAME RECORD
0264	01146	002001		RSS	
0265	01147	027175R		JMP LBERR	INVALID LIBRARY RECORD
0266	01150	016000R		JSB INLST	CHECK IF UNDEFINES PRESENT
0267	01151	016004R		JSB LSTX	
0268	01152	026070R		JMP BLCK6	
0269	01153	163000R		LDA LST4, I	CHECK LST4 FOR -1
0270	01154	050052		CPA N1	
0271	01155	002001		RSS	SKIP--MORE UNDEFINES PRESENT
0272	01156	027151R		JMP --5	LST DEFINED--TRY NEXT LST
0273	01157	063073R		LDA IGNOR	CHECK IF LOADING
0274	01160	002002		SZA	YES
0275	01161	027163R		JMP LISKP	GO SKIP REST OF RECORD
0276	01162	017375R		JSB NAM	PROCESS NAM RECORD
0277	01163	016205R	LISKP	JSB SKIP	SKIP REST OF RECORD
0278	01164	016077R		JSB FETCH	GET NEXT RECORD TYPE
0279	01165	050055	LINNM	CPA .2	ENT RECORD?
0280	01166	027204R		JMP LENTR	YES--SO PROCESS ENT RECORD
0281	01167	050056		CPA .3	DBL RECORD?
0282	01170	027200R		JMP LDBLR	YES--SO PROCESS DBL RECORD
0283	01171	050057		CPA .4	EXT RECORD?
0284	01172	027205R		JMP LEXTR	YES--SO PROCESS EXT RECORD
0285	01173	050060		CPA .5	END RECORD?
0286	01174	027311R		JMP LENDR	YES--SO PROCESS END RECORD
0287	01175	063177R	LBERR	LDA ERR17	
0288	01176	026370R		JMP ABRTX	
0289*					
0290	01177	030467	ERR17	ASC 1, 17	BAD LIBRARY
0291*					
0292	01200	063073R	LDBLR	LDA IGNOR	LOADING?
0293	01201	002003		SZA, RSS	
0294	01202	017533R		JSB DBL	YES - PROCESS DBL RECORD
0295	01203	027163R		JMP LISKP	NO - SKIP REST OF RECORD
0296*					
0297	01204	003401	LENTR	CCA, RSS	
0298	01205	002400	LEXTR	CLA	
0299	01206	073104R		STA NXFLG	SET ENT/EXT FLAG
0300	01207	163035R		LDA CURAL, I	GET # ENTRIES IN ENT/EXT
0301	01210	010071		AND M37	

0302	01211	003004		CMA,INA	
0303	01212	073141R		STA TERMN	SET SYMBOL COUNT
0304	01213	063073R		LDA IGNOR	GET UTILITY PROGRAM LOADING FLAG
0305	01214	002003		SZA,RSS	NOT LOADING
0306	01215	027221R		JMP PRENX	PROCESS ENT/EXT RECORD
0307	01216	063104R		LDA NXFLG	GET ENT/EXT FLAG
0308	01217	002003		SZA,RSS	SKIP=ENT
0309	01220	027163R		JMP LISKP	IGNOR EXT RECORD
0310	01221	016173R	PRENX	JSB BUMP	INCREMENT TO
0311	01222	016173R		JSB BUMP	NAME ADDRESS
0312	01223	163035R	NSYMB	LDA CURAL,I	GET NAME
0313	01224	072210R		STA TBUF	
0314	01225	016173R		JSB BUMP	
0315	01226	163035R		LDA CURAL,I	
0316	01227	072211R		STA TBUF+1	
0317	01230	016173R		JSB BUMP	
0318	01231	163035R		LDA CURAL,I	
0319	01232	072212R		STA TBUF+2	
0320	01233	016173R		JSB BUMP	
0321	01234	017334R		JSB LSCAN	SCAN LST FOR NAME
0322	01235	027252R		JMP ENEX2	NAME NOT IN LST
0323	01236	063073R		LDA IGNOR	GET UTILITY LOADING FLAG
0324	01237	002002		SZA	LOADING
0325	01240	027301R		JMP SCANX	NOT LOADING
0326	01241	063104R		LDA NXFLG	GET ENT/EXT FLAG
0327	01242	002003		SZA,RSS	PROCESS ENT
0328	01243	027262R		JMP ENEX3	SET EXT ORDINAL
0329	01244	163000R		LDA LST4,I	GET ENT/EXT FOR LST ENTRY
0330	01245	002003		SZA,RSS	SKIP=UNDEFINED EXT
0331	01246	027264R		JMP EXNND	IGNORE UTILITY ENTRY POINT
0332	01247	063006R	ENXX1	LDA ID1	
0333	01250	173000R		STA LST4,I	SET LST4=LIBRARY ENT
0334	01251	027264R		JMP EXNND	TEST FOR END OF RECORD
0335	01252	063104R	ENEX2	LDA NXFLG	GET ENT/EXT FLAG
0336	01253	002002		SZA	
0337	01254	027272R		JMP ENEX4	PROCESS ENTRY POINT
0338	01255	017362R		JSB SELST	MAKE LST ENTRY
0339	01256	003400		CCA	
0340	01257	173000R		STA LST4,I	SET LST4 TO UNDEF EXT
0341	01260	002400		CLA	
0342	01261	173001R		STA LST5,I	CLEAR BP LINK ADDRESS WORD
0343	01262	062212R	ENEX3	LDA TBUF+2	
0344	01263	172777R		STA LST3,I	SET ORDINAL INTO LST
0345	01264	063104R	EXNND	LDA NXFLG	GET ENT/EXT FLAG
0346	01265	002002		SZA	EXT
0347	01266	016173R		JSB BUMP	
0348	01267	037141R		ISZ TERMN	SKIP=ALL SYMBOLS PROCESSED
0349	01270	027223R		JMP NSYMB	PROCESS NEXT SYMBOL
0350	01271	027163R		JMP LISKP	GO TO NEXT RECORD
0351	01272	063073R	ENEX4	LDA IGNOR	GET LOADING FLAG
0352	01273	002002		SZA	SKIP=LOADING
0353	01274	027264R		JMP EXNND	IGNORE=ENT NOT IN LIST
0354	01275	017362R		JSB SELST	SET NAME INTO LST
0355	01276	002400		CLA	
0356	01277	173001R		STA LST5,I	CLEAR BP LINK ADDRESS
0357	01300	027247R		JMP ENXX1	CONTINUE PROCESSING ENT POINT

0358	01301	163000R	SCANX	LDA	LST4,I	GET ENT/EXT ENTRY FLAG
0359	01302	002021		SSA,RSS		SKIP=UNDEFINED EXT
0360	01303	027264R		JMP	EXNND	IGNORE DEFINED EXT
0361	01304	063532R		LDA	DSKNM	RESTORE DISC ADDRESS TO START
0362	01305	072074R		STA	BLKAD	OF NAM RECORD
0363	01306	002400		CLA		
0364	01307	073073R		STA	IGNOR	
0365	01310	027143R		JMP	LIRD	GO TO LOAD UTILITY PROGRAM
0366*						
0367	01311	063073R	LENDR	LDA	IGNOR	GET LIBRARY LOADING FLAG
0368	01312	002002		SZA		SKIP=LOADING
0369	01313	027317R		JMP	OMITE	SKIP OVER END RECORD
0370	01314	003400		CCA		SET UTIL LOADING FLAG = IGNOR
0371	01315	073073R		STA	IGNOR	
0372	01316	037065R		ISZ	EXCNT	INCREMENT # UTIL PROG LOADED FLG
0373	01317	016205R	OMITE	JSB	SKIP	SKIP OVER END RECORD
0374	01320	017746R		JSB	UPNAM	UPDATE NAM RECORD IF NECESSARY
0375	01321	016000R		JSB	INLST	INITIALIZE LST
0376	01322	016004R	CLORD	JSB	LSTX	SET LST ADDRESSES
0377	01323	027330R		JMP	END1	TEST FOR DONE
0378	01324	162777R		LDA	LST3,I	
0379	01325	010075		AND	M1774	
0380	01326	172777R		STA	LST3,I	
0381	01327	027322R		JMP	CLORD	GO CLEAR NEXT
0382*						
0383	01330	037333R	END1	ISZ	#LIB	PROCESSED ALL LIB ROUTINES?
0384	01331	027144R		JMP	LIB1	NO= GET NEXT ONE
0385	01332	026070R		JMP	BLCK6	YES= RETURN
0386*						
0387	01333	000000	#LIB	NOP		
0388*						
0389*						
0390*						
0391*						
0392*						
0393*						
0394*						
0395*						
0396*						
0397	01334	000000	LSCAN	NOP		
0398	01335	016000R		JSB	INLST	INITIALIZE LST
0399	01336	016004R	ENTXX	JSB	LSTX	SET LST ADDRESSES
0400	01337	127334R		JMP	LSCAN,I	END OF LST RETURN
0401	01340	062210R		LDA	TBUF	CHECK IF NAME 1,2 SAME?
0402	01341	152775R		CPA	LST1,I	
0403	01342	002001		RSS		YES
0404	01343	027336R		JMP	ENTXX	NO-TRY NEXT ENTRY
0405	01344	062211R		LDA	TBUF+1	CHECK IF NAME 3,4 SAME?
0406	01345	152776R		CPA	LST2,I	
0407	01346	002001		RSS		YES
0408	01347	027336R		JMP	ENTXX	NO-TRY NEXT ENTRY
0409	01350	162777R		LDA	LST3,I	CHECK IF NAME 5 SAME
0410	01351	010075		AND	M1774	
0411	01352	072213R		STA	TBUF+3	
0412	01353	062212R		LDA	TBUF+2	
0413	01354	010075		AND	M1774	

0414	01355	052213R	CPA TBUF+3	
0415	01356	002001	RSS	YES
0416	01357	027336R	JMP ENTXX	NO-TRY NEXT ENTRY
0417	01360	037334R	ISZ LSCAN	INCR EXIT ADDRESS
0418	01361	127334R	JMP LSCAN,I	(N+2)RETURN
0419*				
0420*			SELST SETS CURRENT NAME INTO LST	
0421*				
0422	01362	000000	SELST NOP	
0423	01363	062210R	LDA TBUF	STORE NAME INTO LST
0424	01364	172775R	STA LST1,I	
0425	01365	062211R	LDA TBUF+1	
0426	01366	172776R	STA LST2,I	
0427	01367	062212R	LDA TBUF+2	
0428	01370	010075	AND M1774	
0429	01371	172777R	STA LST3,I	
0430	01372	062773R	LDA TLST	
0431	01373	072774R	STA PLST	SET NEW END OF LST
0432	01374	127362R	JMP SELST,I	RETURN

```

0434*
0435* THIS ROUTINE PROCESSES RELOCATABLE NAM RECORDS.
0436* IT IS ASSUMED THAT "FETCH" HAS BEEN CALLED AND
0437* HAS RETURNED A RECORD TYPE 1. IN ADDITION, THE
0438* BUFFER POINTER IS POINTING TO THE 2ND WORD IN
0439* THE RECORD.
0440*
0441* THREE VARIABLES MUST HAVE PRE-FILLED VALUES:
0442*
0443*     DISCX = "2" FOR SYSTEM DISC
0444*           "3" FOR USER DISC
0445*
0446*     DSKNM = DISC ADDRESS OF NAM RECORD
0447*
0448*     RLGTH = RECORD LENGTH (FROM FETCH ROUTINE)
0449*
0450* FUNCTIONS:
0451*
0452*     1. CHECKS TO SEE IF NAME IS ALREADY IN IDENT
0453*        TABLE
0454*
0455*         A. IF SO, ABORTS WITH ERROR 13.
0456*
0457*         B. IF NOT, CREATES AN IDENT ENTRY AS FOLLOWS:
0458*
0459*             ID1 = CHARACTERS 1&2 OF NAME
0460*             ID2 = CHARACTERS 3&4 OF NAME
0461*             ID3 = CHARACTER 5 OF NAME
0462*             ID4 = COMMON LENGTH (BIT 15 SET IF
0463*                   FROM USER DISC)
0464*             ID5 = DISC ADDRESS
0465*             ID6 = PROGRAM TYPE
0466*             ID7 = ZERO
0467*
0468*     2. CHECKS TO SEE IF PROGRAM HAS ANY BASE PAGE
0469*        ALLOCATED
0470*
0471*         A. IF SO, ABORTS WITH ERROR 14.
0472*
0473*         B. IF NOT, RETURNS NORMALLY.
0474*
0475* 01375 000000 NAM NOP
0476*
0477* GET INFORMATION FROM NAM RECORD
0478*
0479* 01376 016173R JSB BUMP
0480* 01377 016173R JSB BUMP
0481* 01400 163035R LDA CURAL,I NAME.
0482* 01401 073523R STA NAM12 CHARACTERS 1&2
0483* 01402 016173R JSB BUMP
0484* 01403 163035R LDA CURAL,I NAME.
0485* 01404 073524R STA NAM34 CHARACTERS 3&4
0486* 01405 016173R JSB BUMP
0487* 01406 163035R LDA CURAL,I NAME.
0488* 01407 010075 AND M1774
0489* 01410 073525R STA NAM5 CHARACTER 5

```

0490	01411	016173R	JSB BUMP	
0491	01412	163035R	LDA CURAL,I	
0492	01413	073526R	STA NPL	PROGRAM LENGTH
0493	01414	016173R	JSB BUMP	

0546	01467	004010	SLB	
0547	01470	063205R	LDA MSIGN	(BIT 15 SET IF FROM USER DISC)
0548	01471	043530R	ADA NCL	
0549	01472	173011R	STA ID4,I	
0550*				
0551	01473	063532R	LDA DSKNM	DISC ADDRESS OF
0552	01474	173012R	STA ID5,I	NAM RECORD
0553	01475	067526R	LDB NPL	
0554	01476	006021	SSB,RSS	SAVE, IF PROGRAM WAS COMPILED
0555	01477	002400	CLA	
0556	01500	073151R	STA LWH1	
0557*				
0558	01501	067126R	LDB PLGTH	SET PROGRAM TYPE
0559	01502	063531R	LDA NTYPE	
0560	01503	054042	CPB N9	(IF BCS NAM RECORD LENGTH = 9,
0561	01504	060056	LDA .3	FORCE PROGRAM TYPE 3)
0562	01505	010073	AND M177	
0563	01506	002003	SZA,RSS	(IF TYPE 0, FORCE
0564	01507	060056	LDA .3	TYPE 3)
0565	01510	173013R	STA ID6,I	
0566*				
0567	01511	002400	CLA	
0568	01512	173014R	STA ID7,I	
0569	01513	073165R	STA MAXC	INITIALIZE MAX RELOC FOR DBL
0570*				
0571	01514	063003R	LDA TIDNT	INCREMENT IDENT TABLE POINTER
0572	01515	073004R	STA PIDNT	
0573*				
0574	01516	067527R	LDB NBPL	ANY BASE PAGE ALLOCATED?
0575	01517	006003	SZB,RSS	
0576	01520	127375R	JMP NAM,I	NO= RETURN
0577	01521	063200R	LDA ERR14	YES= NOT ALLOWED, ERROR 14 &
0578	01522	026370R	JMP ABRTX	ABORT
0579*				
0580	01523	000000	NAM12 NOP	
0581	01524	000000	NAM34 NOP	
0582	01525	000000	NAM5 NOP	
0583	01526	000000	NPL NOP	
0584	01527	000000	NBPL NOP	
0585	01530	000000	NCL NOP	
0586	01531	000000	NTYPE NOP	
0587	01532	000000	DSKNM NOP	

```

0589*
0590* THIS ROUTINE PROCESSES RELOCATABLE DBL RECORDS.
0591* IT IS ASSUMED THAT "FETCH" HAS BEEN CALLED AND
0592* HAS RETURNED A RECORD TYPE 3. IN ADDITION, THE
0593* BUFFER POINTER IS POINTING TO THE 2ND WORD IN
0594* THE RECORD.
0595*
0596* ONE VARIABLE MUST HAVE A PRE-FILLED VALUE:
0597*
0598*     MAXC  -  MAX RELOCATION ADDRESS IS INITIALLY
0599*             SET TO ZERO IN THE NAM RECORD
0600*             PROCESSOR
0601*
0602* FUNCTION:
0603*
0604*     TO COMPUTE THE LENGTH OF THE CURRENT PROGRAM.
0605*     THIS VALUE IS STORED IN "LWH2" & USED TO
0606*     UPDATE THE NAM RECORD IF THE PROGRAM BEING
0607*     LOADED WAS COMPILED.
0608*
0609 01533 000000 DBL  NOP
0610*
0611 01534 163035R      LDA CURAL,I      SAVE LENGTH OF CURRENT RECORD
0612 01535 010072      AND M77
0613 01536 073147R      STA TEMP1
0614*
0615 01537 016173R      JSB BUMP
0616 01540 016173R      JSB BUMP
0617 01541 167035R      LDB CURAL,I      GET CURRENT RELOCATION ADDRESS
0618 01542 063165R      LDA MAXC        IS IT GREATER THAN CURRENT MAX
0619 01543 003004      CMA,INA          RELOCATION ADDRESS?
0620 01544 040001      ADA B
0621 01545 002020      SSA
0622 01546 127533R      JMP DBL,I        NO- SKIP IT
0623*
0624 01547 063147R      LDA TEMP1        YES- COMPUTE NEW PROGRAM
0625 01550 040001      ADA B              LENGTH
0626 01551 073152R      STA LWH2
0627 01552 077165R      STB MAXC        SET NEW MAX RELOC ADDR
0628 01553 127533R      JMP DBL,I

```

```

0630*
0631* THIS ROUTINE PROCESSES RELOCATABLE ENT & EXT
0632* RECORDS. IT IS ASSUMED THAT "FETCH" HAS BEEN
0633* CALLED AND HAS RETURNED RECORD TYPE 2 OR 4. IN
0634* ADDITION, THE BUFFER POINTER IS POINTING TO THE
0635* SECOND WORD IN THE BUFFER.
0636*
0637* PRE-FILLED VALUES:
0638*
0639*     "A" REGISTER - 0   IF EXT RECORD
0640*                  -1   IF ENT RECORD
0641*
0642* FUNCTIONS:
0643*
0644*     1. CHECKS TO SEE IF ENT/EXT NAMES ARE IN LST
0645*
0646*         A. YES:   ENT: IF LST4>0, ERROR 7 (DUP ENT)
0647*                      IF LST4=0, SET LST4 TO ID1
0648*                      IF LST4<0, ERROR 15 (SEG
0649*                          BEFORE MAIN) IF MAIN;
0650*                          OTHERWISE SET LST4 TO
0651*                          ID1
0652*                      EXT: IF MAIN, CONTINUE PROCESS
0653*                          IF SEGMENT & LST4<=0, CONT.
0654*                          IF SEGMENT & LST4>0, FIND ID
0655*                          ENTRY OF ENT NAME. IF MAIN
0656*                          SET ID1 TO ID1 OF DEFINING
0657*                          PROGRAM.
0658*
0659*         B. NO:   ENT: PUT IN LST, SET LST4 TO ID1
0660*
0661*                      EXT: PUT IN LST, SET LST4 TO 0
0662*                          IF NOT SEGMENT; TO COMPLE-
0663*                          MENT OF ID1 IF SEGMENT
0664*
0665  01554 000000  EN.XT NOP
0666  01555 073104R      STA NXFLG      SAVE ENT/EXT FLAG
0667*
0668  01556 163035R      LDA CURAL,I     SET COUNTER TO - # NAMES
0669  01557 010071      AND M37         IN RECORD
0670  01560 003004      CMA,INA
0671  01561 073065R      STA EXCNT
0672*
0673  01562 016173R      JSB BUMP        BUMP BUFFER POINTER TO START OF
0674  01563 016173R      JSB BUMP        SYMBOL NAME
0675*
0676  01564 163035R  SAVNM LDA CURAL,I     SAVE SYMBOL NAME IN TBUF
0677  01565 072210R      STA TBUF
0678  01566 016173R      JSB BUMP
0679  01567 163035R      LDA CURAL,I
0680  01570 072211R      STA TBUF+1
0681  01571 016173R      JSB BUMP
0682  01572 163035R      LDA CURAL,I
0683  01573 010075      AND M1774
0684  01574 072212R      STA TBUF+2
0685  01575 016173R      JSB BUMP        POSITION POINTER TO NEXT NAME

```

```

0686*
0687 01576 016000R JSB INLST SEARCH LST FOR NAME
0688 01577 016004R NEXT JSB LSTX
0689 01600 027646R JMP NOENT - NOT FOUND
0690 01601 062210R LDA TBUF
0691 01602 152775R CPA LST1,I
0692 01603 002001 RSS
0693 01604 027577R JMP NEXT
0694 01605 062211R LDA TBUF+1
0695 01606 152776R CPA LST2,I
0696 01607 002001 RSS
0697 01610 027577R JMP NEXT
0698 01611 162777R LDA LST3,I
0699 01612 010075 AND M1774
0700 01613 052212R CPA TBUF+2
0701 01614 002001 RSS
0702 01615 027577R JMP NEXT
0703*
0704* NAME FOUND IN LST
0705*
0706 01616 063104R LDA NXFLG ENT OR EXT?
0707 01617 002003 SZA,RSS
0708 01620 027672R JMP EXT1 EXT PROCESSING
0709*
0710 01621 163000R LDA LST4,I ID ADDRESS ALREADY ASSIGNED?
0711 01622 002003 SZA,RSS
0801 01623 027643R JMP SIDA NO- SET ID ADDRESS
0802 01624 002021 SSA,RSS
0803 01625 027635R JMP DUP YES- DUP ENTRY POINT ERROR
0804 01626 163013R LDA ID6,I
0805 01627 010073 AND M177
0806 01630 050056 CPA .3
0807 01631 002001 RSS
0808 01632 027643R JMP SIDA
0809*
0810 01633 063201R LDA ERR15
0811 01634 026370R JMP ABRTX
0812*
0813 01635 063171R DUP LDA ERR07 DUPLICATE ENTRY POINT ERROR
0814 01636 016114R JSB ERROR
0815 01637 060060 LDA .5 OUTPUT NAME, THEN ABORT
0816 01640 066775R LDB LST1
0817 01641 016101R JSB SYOUT
0818 01642 026371R JMP ABORT
0819*
0820 01643 063006R SIDA LDA ID1 SET ADDRESS OF ID ENTRY INTO
0821 01644 173000R STA LST4,I LST4
0822 01645 027730R JMP ENXT1
0823*
0824* NAME NOT FOUND IN LST
0825*
0826 01646 062210R NOENT LDA TBUF PUT SYMBOL NAME IN LST
0827 01647 172775R STA LST1,I
0828 01650 062211R LDA TBUF+1
0829 01651 172776R STA LST2,I
0830 01652 062212R LDA TBUF+2

```

0031	01653	172777R	STA LST3,I	
0032	01654	002400	CLA	CLEAR BP LINK IN LST
0033	01655	173001R	STA LST5,I	
0034	01656	062773R	LDA TLST	INCREMENT LST POINTER
0035	01657	072774R	STA PLST	
0036*				
0037	01660	063104R	LDA NXFLG	SET LST4 ACCORDING TO RECORD
0038	01661	002002	SZA	TYPE:
0039	01662	027643R	JMP SIDA	IF ENT, LST4 = ID1
0040	01663	163013R	LDA ID6,I	IF EXT, LST4 = -ID1 IF SEGMENT
0041	01664	010073	AND M177	= 0 IF MAIN
0042	01665	067006R	LDB ID1	
0043	01666	050060	CPA .5	
0044	01667	007001	CMB,RSS	
0045	01670	006400	CLB	
0046	01671	177000R	STB LST4,I	
0047*				
0048	01672	163013R EXT1	LDA ID6,I	FURTHER PROCESSING NECESSARY IF
0049	01673	010073	AND M177	PROGRAM CONTAINING EXT IS A
0050	01674	050060	CPA .5	SEGMENT
0051	01675	002001	RSS	
0052	01676	027730R	JMP ENXT1	OTHERWISE. ON TO THE NEXT
0053*				
0054	01677	063006R	LDA ID1	SAVE ID ADDRESS OF SEGMENT
0055	01700	073074R	STA IMAIN	
0056*				
0057	01701	163000R	LDA LST4,I	HAS ERRORESPONDING ENTRY BEEN
0058	01702	002002	SZA	DEFINED?
0059	01703	002020	SSA	
0060	01704	027730R	JMP ENXT1	NO- SKIP
0061	01705	073003R	STA TIDNT	YES- SET TO GET ID'S FOR PROG
0062	01706	016035R	JSB IDX	DEFINING ENT
0063	01707	026637R	JMP RCERR	(SHOULD FIND IT)
0064	01710	163013R	LDA ID6,I	IS PROGRAM CONTAINING ENTRY POIN
0065	01711	002021	SSA,RSS	A SUBROUTINE?
0066	01712	027716R	JMP NTMN	YES- SET FOR NOT MAIN
0067	01713	010073	AND M177	NO - IS IT A MAIN?
0068	01714	050056	CPA .3	
0069	01715	007401	CCB,RSS	YES, TEMP=-1 FOR MAIN
0070	01716	006400 NTMN	CLB	
0071	01717	077140R	STB TEMP	
0072*				
0073	01720	063074R	LDA IMAIN	RESTORE IDENT ENTRY OF PROGRAM
0074	01721	073003R	STA TIDNT	CONTAINING EXT.
0075*				
0076	01722	016035R	JSB IDX	
0077	01723	026637R	JMP RCERR	
0078*				
0079	01724	037140R	ISZ TEMP	WAS PROGRAM CONTAINING ENT A MAI
0080	01725	027730R	JMP ENXT1	NO- GET NEXT SYMBOL
0081	01726	163000R	LDA LST4,I	YES- SET ID ADDRESS OF MAIN
0082	01727	173014R	STA ID7,I	INTO ID7 OF SEGMENT
0083*				
0084	01730	063104R ENXT1	LDA NXFLG	IF ENT, BUMP POINTER PAST
0085	01731	002002	SZA	RELOCATABLE ADDRESS
0086	01732	016173R	JSB BUMP	

0087	01733	037065R	ISZ EXCNT	DONE WITH ALL SYMBOLS IN RECORD?
0088	01734	027564R	JMP SAVNM	NO- PROCESS NEXT NAME
0089	01735	127554R	JMP EN.XT,I	YES- RETURN

```

0091*
0092* THIS ROUTINE PROCESSES RELOCATABLE END RECORDS.
0093* IT IS ASSUMED THAT "FETCH" HAS BEEN CALLED AND
0094* HAS RETURNED A RECORD TYPE OF 5. IN ADDITION,
0095* THE BUFFER POINTER IS POINTING TO THE 2ND WORD
0096* IN THE RECORD.
0097*
0098* TWO VARIABLES MUST HAVE PREFILLED VALUES:
0099*
0100*     LWH1 = "0" IF PROGRAM NOT COMPILED
0101*           "DISC ADDR OF NAM RECORD" IF PROGRAM
0102*           WAS COMPILED
0103*           (SET BY NAM RECORD PROCESSOR)
0104*
0105*     LWH2 = PROGRAM LENGTH
0106*           (SET BY DBL RECORD PROCESSOR)
0107*
0108* FUNCTIONS:
0109*
0110*     1. TO SET BIT 15 OF ID6 FOR THIS PROGRAM
0111*
0112*         "0" - PROGRAM IS A SUBROUTINE (NO LABEL
0113*         ON END STATEMENT)
0114*
0115*         "1" - PROGRAM IS A MAIN/SEGMENT (LABEL
0116*         ON END STATEMENT)
0117*
0118*     2. TO SET THE COMPUTED RECORD LENGTH INTO THE
0119*     NAM RECORD IF THE PROGRAM WAS COMPILED.
0120*     (THIS IS NECESSARY FOR CURRENT PAGE LINK
0121*     PROCESSING.)
0122*
0123* 01736 000000 END NOP
0124*
0125* 01737 002404 CLA,INA SET BIT 15 OF ID6 IF PROGRAM
0126* 01740 113035R AND CURAL,I HAD A LABEL ON THE END
0127* 01741 001300 RAR STATEMENT (IE. WAS A MAIN OR
0128* 01742 133013R IOR ID6,I A SEGMENT)
0129* 01743 173013R STA ID6,I
0130* 01744 017746R JSB UPNAM UPDATE NAM RECORD IF NECESSARY
0131*
0132* 01745 127736R JMP END,I
0133*
0134* UPDATE NAM RECORD = PROG LENGTH & CHECKSUM
0135*
0136* 01746 000000 UPNAM NOP
0137* 01747 063151R LDA LWH1 WAS PROGRAM COMPILED?
0138* 01750 002003 SZA,RSS
0139* 01751 127746R JMP UPNAM,I NO-RETURN
0140*
0141* 01752 066265R LDB #WRDS SAVE CURRENT # WORDS FOR
0142* 01753 077147R STB TEMP1 DISC TRANSFER
0143* 01754 067273R LDB .128 SET TO INPUT 1 SECTOR
0144* 01755 076265R STB #WRDS
0145*
0146* 01756 067274R LDB ADBUF INPUT NAM RECORD INTO DBUF

```

0147	01757	016235R	JSB DISKI	
0148	01760	063152R	LDA LWH2	SET IN COMPUTED PROGRAM
0149	01761	033205R	IOR MSIGN	LENGTH
0150	01762	073334R	STA DBUF+6	
0151	01763	063326R	LDA DBUF	COMPUTE NEW CHECKSUM
0152	01764	001727	ALF, ALF	
0153	01765	010073	AND M177	
0154	01766	003004	CMA, INA	
0155	01767	040056	ADA .3	
0156	01770	073150R	STA WDCNT	
0157	01771	063327R	LDA DBUF+1	
0158	01772	067274R	LDB ADBUF	
0159	01773	044056	ADB .3	
0160	01774	140001	ADA B, I	
0161	01775	006004	INB	
0162	01776	037150R	ISZ WDCNT	
0163	01777	027774R	JMP *-3	
0164	02000	073330R	STA DBUF+2	
0165*				
0166	02001	063151R	LDA LWH1	OUTPUT UPDATED NAM RECORD
0167	02002	067274R	LDB ADBUF	BACK TO DISC
0168	02003	016243R	JSB DISKO	
0169*				
0170	02004	063147R	LDA TEMP1	RESTOR # WORDS
0171	02005	072265R	STA #WRDS	FOR DISC TRANSFER
0172*				
0173	02006	127746R	JMP UPNAM, I	

```

0175*
0176* SETUP - SETS UP THE RETURN ADDRESS FOR BLOCK
0177*          ROUTINE WHEN ALL SECTORS HAVE BEEN INPUT
0178*          SETS UP THE BLOCK SIZE (BOTH # SECTORS &
0179*          # WORDS) FOR DISC INPUT, ACCORDING TO
0180*          TDIR.
0181*
0182* DEPENDENT VARIABLES:
0183*
0184*          "A" REGISTER - RETURN ADDRESS WHEN ALL
0185*                      SECTORS HAVE BEEN INPUT
0186*
0187*          TDIR          = IF JBIN (=1), BLOCK IS 1
0188*                      = IF NOT JBIN (=0), BLOCK
0189*                      IS 4 SECTORS
0190*
0191 02007 000000 SETUP NOP
0192 02010 072073R STA RTRN SET RETURN FOLLOWING COMPLETION
0193*
0194 02011 002404 CLA,INA IF INPUT IS FROM JOB BINARY AREA
0195 02012 067137R LDB TDIR BLOCK SIZE IS 1 SECTOR.
0196 02013 006003 SZB,RSS OTHERWISE, BLOCK SIZE IS 4
0197 02014 040056 ADA ,3 SECTORS.
0198 02015 072075R STA BLCK#
0199 02016 016003X MPY ,128 ALTER # WORDS PER DISC INPUT
0200 02020 072265R STA #WRDS
0201*
0202 02021 126007R JMP SETUP,I

```

```

0204*
0205* BLOCK - INPUTS A BLOCK OF RELOCATABLE RECORDS
0206*          FROM THE DISC INTO CORE AT "LBUF". THE
0207*          SIZE OF THE BLOCK IS DETERMINED BY
0208*          "SETUP". IF THE ENTIRE BLOCK CANNOT BE
0209*          INPUT, AS MUCH AS POSSIBLE IS. IF NONE
0210*          IS POSSIBLE, RETURN IS MADE THROUGH
0211*          "RTRN" (SET BY "SETUP"), AFTER RESTORING
0212*          THE NORMAL DISC INPUT TRANSFER TO 1
0213*          SECTOR (128 WORDS).
0214*          FOLLOWING EACH TRANSFER, THE BUFFER
0215*          POINTER & LENGTH ("CURAL" & "LCNT") ARE
0216*          INITIALIZED; # SECTORS REMAINING IS
0217*          DECREMENTED; DISC ADDRESS & # SECTORS
0218*          INPUT ARE INCREMENTED.
0219*

```

```

0220* DEPENDENT VARIABLES:
0221*
0222*          RTRN - RETURN ADDRESS FOLLOWING COMPLETION
0223*                (SET BY SETUP)
0224*
0225*          DISCX= "2" FOR SYSTEM DISC
0226*                "3" FOR USER DISC
0227*
0228*          TDIR = "0" FOR NON-JBIN FILES

```

```

0229*          "1" FOR JBIN FILES (BACKWARD INCR)
0230*
0231*          COUNT= # SECTORS TO BE INPUT
0232*
0233*          BLCK#= BLOCK SIZE OF INPUT TRANSFER
0234*
0235*          BCKAD= STARTING DISC ADDRESS
0236*
0237 02022 000000 BLOCK NOP
0238*
0239 02023 062074R      LDA BLKAD      SAVE CURRENT BLOCK
0240 02024 072076R      STA CBLA      STARTING ADDRESS
0241*
0242 02025 066075R      LDB BLCK#      CAN INPUT DESIRED BLOCK SIZE?
0243 02026 007004      CMB,INB
0244 02027 047032R      ADB COUNT
0245 02030 006020      SSB
0246 02031 026057R      JMP BLCK1      NO= SEE IF ANY INPUT POSSIBLE
0247*
0248 02032 077032R BLCK2 STB COUNT      UPDATE # SECTORS LEFT TO INPUT
0249 02033 062074R      LDA BLKAD
0250 02034 067276R      LDB ALBUF      INPUT BLOCK FROM DISC ADDR IN
0251 02035 016235R      JSB DISKI      BLKAD INTO LBUF
0252*
0253 02036 063276R      LDA ALBUF      INITIALIZE BUFFER POINTER
0254 02037 073035R      STA CURAL
0255 02040 062075R      LDA BLCK#      INITIALIZE BUFFER LENGTH
0256 02041 016003X      MPY .128      ( = # WORDS)
0257 02043 003004      CMA,INA
0258 02044 073077R      STA LCNT
0259*
0260 02045 062076R      LDA BLCK#
0261 02046 003004      CMA,INA
0262 02047 073140R      STA TEMP
0263 02050 037063R BLCK5 ISZ DSCNT      INCREMENT # SECTORS INPUT
0264 02051 062074R      LDA BLKAD      INCREMENT DISC ADDRESS
0265 02052 017204R      JSB NDA
0266 02053 072074R BLCK3 STA BLKAD
0267 02054 037140R      ISZ TEMP
0268 02055 026050R      JMP BLCK5
0269 02056 126022R      JMP BLOCK,I
0270*
0271 02057 063032R BLCK1 LDA COUNT      ANY MORE SECTORS TO INPUT?
0272 02060 002003      SZA,RSS
0273 02061 026070R      JMP BLCK6      NO= RETURN
0274 02062 072075R      STA BLCK#      YES= SET BLCK# TO # SECTORS
0275 02063 016003X      MPY .128
0276 02065 072265R      STA #WRDS
0277 02066 006400      CLB
0278 02067 026032R      JMP BLCK2      LEFT; SET # SECTORS LEFT
                                         TO ZERO.
0279*
0280 02070 063273R BLCK6 LDA .128      RESTORE # WORDS PER DISC INPUT
0281 02071 072265R      STA #WRDS      TRANSFER TO 128
0282 02072 126073R      JMP RTRN,I    RETURN
0283*
0284 02073 000000 RTRN  NOP      RETURN ADDRESS

```

PAGE 0037 #03 UTILITIES FOR RELOCATABLE DISC INPUT(PASS 1)

0285	02074	000000	BLKAD	NOP	STARTING DISC ADDRESS
0286	02075	000000	BLCK#	NOP	BLOCK SIZE
0287	02076	000000	CBLA	NOP	CURRENT BLCK DISC ADDRESS

0289*
 0290* NOTE: FETCH, BUMP, & SKIP ROUTINES ARE ALSO
 0291* USED WHEN PROCESSING RELOCATABLE INPUT
 0292* FROM PERIPHERAL DEVICES. IN THESE CASES,
 0293* IT IS ASSUMED THAT NO RECORD LENGTH WILL
 0294* EXCEED 128 WORDS.

0296*
 0297* FETCH - SETS UP THE RECORD LENGTH OF THE CURRENT
 0298* RELOCATABLE RECORD (RLGTH & SCNT)
 0299* VERIFIES RECORD VALIDITY:
 0300* IF ILLEGAL LENGTH OR TYPE = ERROR 2

0301*
 0302* NOTE: IF A ZERO RECORD LENGTH IS EN-
 0303* COUNTERED, THE RECORD IS ASSUMED TO
 0304* CONTAIN THE ZEROES FOLLOWING THE END
 0305* RECORD. THE REST OF THE SECTOR IS
 0306* SKIPPED

0307*
 0308* DEPENDENT VARIABLES:

0309*
 0310* CURAL - ASSUMED TO BE POINTING TO 1ST
 0311* WORD IN RELOCATABLE RECORD ON
 0312* ENTRY. ON EXIT, IT IS POINTING
 0313* TO THE 2ND WORD.

0314*
 0315 02077 000000 FETCH NOP

0316*
 0317 02100 163035R LDA CURAL,I
 0318 02101 002020 SSA
 0319 02102 026637R JMP RCERR ILLEGAL RECORD LENGTH
 0320 02103 001727 ALF,ALF
 0321 02104 010073 AND M177
 0322 02105 003004 CMA,INA
 0323 02106 073126R STA RLGTH SET VALID RECORD LENGTH
 0324*
 0325 02107 002003 SZA,RSS RECORD LENGTH=0?
 0326 02110 026161R JMP NEWS YES- GET NEXT SECTOR
 0327 02111 072172R STA SCNT NO- SET RECORD LENGTH FOR SKIP
 0328*
 0329 02112 040056 ADA .3 VALID RECORD LENGTH IS
 0330 02113 002021 SSA,RSS 4 - 60
 0331 02114 026637R JMP RCERR
 0332 02115 073150R STA WDCNT
 0333 02116 043271R ADA D57
 0334 02117 002020 SSA
 0335 02120 026637R JMP RCERR
 0336*
 0337 02121 016173R JSB BUMP BUMP POINTER TO 2ND WORD
 0338*
 0339 02122 163035R CHKOK LDA CURAL,I GET RECORD TYPE
 0340 02123 001723 ALF,RAR VALID IF 1-5
 0341 02124 010062 AND .7

0342	02125	070001	STA B	
0343	02126	002002	SZA	
0344	02127	044045	ADB N6	
0345	02130	006021	SSB,RSS	
0346	02131	026637R	JMP RCERR	NOT VALID - ERROR 2
0347*				
0348	02132	050054	CPA .1	NAM RECORD?
0349	02133	002001	RSS	YES- UPDATE DISC ADDR
0350	02134	126077R	JMP FETCH,I	CONTAINS RECORD TYPE
0351*				
0352	02135	007400	CCB	COMPUTE NEW DISC ADDRESS
0353	02136	063077R	LDA LCNT	FOR NAM RECORD
0354	02137	040052	ADA N1	
0355	02140	016002X	DIV N128	
0356	02142	066075R	LDB BLCK#	
0357	02143	007004	CMB,INB	
0358	02144	040001	ADA B	
0359	02145	002021	SSA,RSS	
0360	02146	026157R	JMP NEW2	
0361	02147	073136R	STA TCNT	
0362*				
0363	02150	062076R	LDA CBLA	
0364	02151	017204R	JSB NDA	
0365	02152	037136R	ISZ TCNT	
0366	02153	026151R	JMP *-2	
0367	02154	073532R	STA DSKNM	
0368	02155	002404	CLA,INA	"A" = NAM RECORD TYPE
0369	02156	126077R	JMP FETCH,I	
0370*				
0371	02157	062076R	LDA CBLA	
0372	02160	026154R	JMP NEW1	
0373*				
0374	02161	007400	CCB	POSITION POINTERS TO
0375	02162	063077R	LDA LCNT	BEGINNING OF NEXT
0376	02163	016002X	DIV .128	SECTOR
0377	02165	006003	SZB,RSS	
0378	02166	067272R	LDB N128	
0379	02167	076172R	STB SCNT	
0380	02170	016205R	JSB SKIP	
0381	02171	026100R	JMP FETCH+1	REPROCESS NAM RECORD
0382	02172	000000	SCNT NOP	

0384*
 0385* BUMP - INCREMENTS BUFFER ADDRESS POINTER, 2
 0386* DECREMENTS # WORDS WITHIN RELOCATABLE
 0387* RECORD,
 0388* DECREMENTS # WORDS IN BUFFER
 0389* WHEN ALL WORDS HAVE BEEN PROCESSED, "BLOCK"
 0390* IS CALLED TO READ IN NEXT BLOCK OF
 0391* SECTORS

0392*
 0393* DEPENDENT VARIABLES:
 0394*
 0395* CURAL - CURRENT BUFFER ADDRESS POINTER

```

0396*
0397*      LCNT  = # WORDS REMAINING IN BUFFER
0398*
0399*      SCNT  = # WORDS REMAINING IN RECORD
0400*
0401  02173 000000 BUMP  NOP
0402*
0403  02174 037035R      ISZ CURAL      INCREMENT ADDR POINTER
0404  02175 036510R      ISZ W,PRG      INCREMENT # WORDS IN PROGRAM
0405  02176 000000      NOP              (IGNORE ZERO)
0406  02177 036172R      ISZ SCNT       INCREMENT # WORDS IN RECORD
0407  02200 000000      NOP              (IGNORE ZERO)
0408  02201 037077R      ISZ LCNT       INCREMENT # WORDS IN BUFFER
0409  02202 126173R      JMP BUMP,I      NEW BLOCK NOT NEEDED
0410*
0411  02203 016022R      JSB BLOCK       INPUT NEW BLOCK
0412  02204 126173R      JMP BUMP,I

0414*
0415*  SKIP =  SKIPS TO THE BEGINNING OF THE NEXT
0416*          RELOCATABLE RECORD IN THE INPUT BUFFER
0417*          (DOES NOT SKIP IF ALREADY POSITIONED TO
0418*          BEGINNING OF RECORD = IE SCNT#0)
0419*
0420*      DEPENDENT VARIABLES:
0421*
0422*          SCNT  = # WORDS REMAINING IN RECORD
0423*                (SET BY "FETCH", UPDATED BY "BUMP")
0424*
0425  02205 000000 SKIP  NOP
0426*
0427  02206 062172R      LDA SCNT        ALREADY AT BEGINNING OF RECORD?
0428  02207 002021      SSA,RSS
0429  02210 126205R      JMP SKIP,I      YES= RETURN
0430*
0431  02211 073136R      STA TCNT        NO = BUMP POINTERS PAST END
0432  02212 016173R      JSB BUMP        OF CURRENT RECORD
0433  02213 037136R      ISZ TCNT
0434  02214 026212R      JMP *-2
0435  02215 126205R      JMP SKIP,I

```

```

0437*
0438*  VERIFY THAT NO DUPLICATE FILE NAMES EXIST
0439*
0440  02216 002400 LMAIN CLA
0441  02217 073121R STA PTYPE
0442  02220 016031R JSB INIDX CLEAR PROGRAM USAGE FLAGS
0443  02221 016035R CLRID JSB IDX SET CURRENT IDENT ADDRESSES
0444  02222 026267R JMP IDCLR
0445  02223 163010R LDA ID3,I
0446  02224 010075 AND M1774
0447  02225 173010R STA ID3,I CLEAR USAGE FLAG
0448  02226 163013R LDA ID6,I CHECK PROGRAM TYPE
0449  02227 010073 AND M177
0450  02230 050056 CPA .3 CHECK IF MAIN
0451  02231 026244R JMP USCHK
0452  02232 050057 CPA .4 CHECK IF DEVICE DRIVER
0453  02233 026265R JMP DDRV
0454  02234 050060 CPA .5 CHECK IF SEGMENT
0455  02235 026244R JMP USCHK
0456  02236 002002 SZA CHECK IF TYPE 0
0457  02237 026221R JMP CLRID NO, SO NOT MAIN
0458  02240 163013R CLR0 LDA ID6,I CONVERT TYPES 0 & 4 TO 3
0459  02241 010075 AND M1774
0460  02242 030056 IOR .3
0461  02243 173013R STA ID6,I
0462  02244 063006R USCHK LDA ID1
0463  02245 072251R STA ++4
0464  02246 016001X JSB EXEC
0465  02247 002253R DEF ++4
0466  02250 007174R DEF .18 REQUEST CODE
0467  02251 002251R DEF * FILE NAME ADDRESS
0468  02252 005202R DEF SECT # SECTORS
0469  02253 063202R LDA SECT GET # SECTORS IN FILE
0470  02254 002003 SZA,RSS NON-ZERO--ALREADY IN FILE
0471  02255 026221R JMP CLRID TRY NEXT PROGRAM
0472  02256 063170R LDA ERR06 PRINT DUPL FILE NAME ERROR CODE
0473  02257 016114R JSB ERROR
0474  02260 060060 LDA .5 PRINT FILE NAME
0475  02261 067006R LOB ID1
0476  02262 016101R JSB SYOUT
0477  02263 037121R ISZ PTYPE INCR DUPL FILE FLAG
0478  02264 026221R JMP CLRID SEE IF ANY MORE DUPL FILES
0479*
0480  02265 073156R DDRV STA DVR SET DRIVER FLAG # 0
0481  02266 026240R JMP CLR0 SET TYPE TO 3 TEMPORARILY
0482  02267 063121R IDCLR LDA PTYPE CHECK IF ANY DUPL FILE FLAG SET
0483  02270 002002 SZA NO
0484  02271 026371R JMP ABORT YES--SO ABORT LOADER
0485*
0486*  INITIALIZE DIRECTORY
0487*
0488  02272 060200 LDA DISCL DISCL+1 * START T/S OF USER DIR
0489  02273 002004 INA
0490  02274 073053R STA DSKID
0491  02275 063275R DRSCH LDA AIBUF SET INITIAL IBUF POINTER
0492  02276 073034R STA CURAI

```

0493	02277	063272R	LDA N120	SET IBUF COUNT
0494	02300	073071R	STA ICNT	
0495	02301	017537R	JSB SETCR	SET USER DISC FLAG IN BASE PAGE
0496	02302	063053R	LDA DSKID	A IS DISC ADDRESS
0497	02303	067275R	LDB AIBUF	B IS CORE ADDRESS
0498	02304	016235R	JSB DISKI	READ IN 1 SECTOR OF USER DIR
0499	02305	163034R	DRCHK LDA CURAI, I	
0500	02306	002003	SZA, RSS	CHECK IF AT END OF DIRECTORY
0501	02307	026347R	JMP ENDDR	END OF USER DIR FOUND
0502	02310	050052	CPA N1	CHECK IF AT END OF DIR SECTOR
0503	02311	002001	RSS	END OF SECTOR
0504	02312	026322R	JMP DRCH1	NOT AT END OF SECTOR
0505	02313	060174	DRCH0 LDA TSONE	INCREMENT SECTOR ADDRESS
0506	02314	073053R	STA DSKID	
0507	02315	010073	AND M177	
0508	02316	002002	SZA	
0509	02317	026275R	JMP DRSCH	GO READ IN NEXT SECTOR
0510	02320	063174R	DRCHA LDA ERR10	
0511	02321	026370R	JMP ABRTX	OUT OF DIRECTORY SPACE
0512	02322	067034R	DRCH1 LDB CURAI	POINTER IN IBUF
0513	02323	044055	ADB .2	INCR TO NAME 5 - TYPE ADDRESS
0514	02324	160001	LDA B, I	
0515	02325	010073	AND M177	ISOLATE TYPE
0516	02326	040045	ADA N6	CHECK IF 5 OR 11 WORD ENTRY
0517	02327	002021	SSA, RSS	SKIP=11 WORD DIRECTORY ENTRY
0518	02330	026337R	JMP DRCH5	5 WORD DIRECTORY ENTRY
0519	02331	044064	ADB .9	PROCESS 11 WORD ENTRY
0520	02332	077034R	STB CURAI	
0521	02333	067071R	LDB ICNT	
0522	02334	047170R	ADB .11	
0523	02335	077071R	STB ICNT	
0524	02336	026344R	JMP DRCH6	GO TRY NEXT ENTRY
0525	02337	044056	DRCH5 ADB .3	PROCESS 5 WORD ENTRY
0526	02340	077034R	STB CURAI	
0527	02341	067071R	LDB ICNT	
0528	02342	044060	ADB .5	
0529	02343	077071R	STB ICNT	
0530	02344	006021	DRCH6 SSB, RSS	CHECK IF PAST END OF SECTOR
0531	02345	026313R	JMP DRCH0	
0532	02346	026305R	JMP DRCHK	TRY NEXT ENTRY
0533	02347	063071R	ENDDR LDA ICNT	CHECK IF ROOM FOR MORE ENTRIES
0534	02350	040065	ADA .10	
0535	02351	002020	SSA	NO=SKIP
0536	02352	026370R	JMP LDAB5	YES
0537	02353	003400	CCA	
0538	02354	017421R	JSB OUTID	OUTPUT MORE DIRECTORY FOLLOWS FLA
0539	02355	067071R	LDB ICNT	
0540	02356	057272R	CPB N120	CHECK IF NEW SECTOR
0541	02357	026365R	JMP DRCLR	GO CLEAR NEXT DIRTRY SECTOR
0542	02360	077140R	STB TEMP	SAVE ZERO OUTPUT COUNT
0543	02361	002400	CLA	
0544	02362	017421R	JSB OUTID	OUTPUT ZERO TO DIRECT. BUFFER
0545	02363	037140R	ISZ TEMP	
0546	02364	026361R	JMP *-3	GO OUTPUT ANOTHER ZERO
0547	02365	063053R	DRCLR LDA DSKID	
0548	02366	067275R	LDB AIBUF	

Address	Instruction	Comment
0549	02367 016243R JSB DISK0	CLEAR NEXT DRCTRY SECTOR
0550*		
0551*	LOAD RELOCATABLES & CREATE ABSOLUTE CODE	
0552*		
0553	02370 063154R LDABS LDA UMFWA	SET INITIAL PROG RELOC ADDRESS
0554	02371 073025R STA BGBND	
0555	02372 073325R STA COMAD	
0556	02373 073123R STA RELAD	
0557	02374 073117R STA PPREL	
0558	02375 063157R LDA LISTU	
0559	02376 010072 AND M77	
0560	02377 033200R IOR M1100	
0561	02400 073140R STA TEMP	
0562	02401 016001X JSB EXEC	CALL EXEC TO PAGE LIST DEVICE
0563	02402 002406R DEF **4	
0564	02403 000056 DEF .3	
0565	02404 011140R DEF TEMP	
0566	02405 000041 DEF N10	
0567	02406 063174R PRHED LDA .18	PRINT -RELOCATING LOADER- HEADING
0568	02407 067236R LDB D0SHD	
0569	02410 016134R JSB DRKEY	
0570	02411 016125R JSB SPACE	
0571	02412 063173R LDA .16	PRINT -NAME/ENTRY ADDR - HEADING
0572	02413 067250R LDB NAENT	
0573	02414 016134R JSB DRKEY	
0574	02415 016125R JSB SPACE	
0575	02416 063272R LDA N128	
0576	02417 073044R STA DCNT	INITIALIZE ABSOLUTE OUTPUT COUNT
0577	02420 067274R LDB ADBUF	
0578	02421 077033R STB CURAD	SET CURRENT DBUF ADDRESS
0579	02422 063047R LDA DSKAB	
0580	02423 073050R STA DSKAD	SET CURRENT ABS DISC ADDRESS
0581	02424 060056 LDA .3	SET PROGRAM TYPE TO USER MAIN
0582	02425 073121R STA PTYPE	
0583	02426 063117R LDA PPREL	
0584	02427 073151R STA XPREL	
0585	02430 063002R LDA BIDNT	
0586	02431 073005R STA CIDNT	SET CIDNT FOR IDSCN
0587	02432 063304R LDA PLAD	
0588	02433 073036R STA CURAP	
0589	02434 002400 CLA	
0590	02435 073346R STA LIBRY	
0591	02436 073111R STA PGCNT	
0592	02437 037100R ISZ LDING	SET LOADING FLAG
0593	02440 016210R BDLD JSB IDSCN	SCAN FOR USER MAIN
0594	02441 026600R JMP BDEND	END OF USER MAIN
0595	02442 167010R LDB ID3,I	GET USAGE FLAG
0596	02443 006014 SLB,INB	SKIP-PROG NOT LOADED
0597	02444 026440R JMP BDLD	PROG LOADED
0598	02445 177010R STB ID3,I	SET PROGRAM LOADED FLAG
0599	02446 163011R LDA ID4,I	WD 4 OF IDENT TABLE FOR COMMON
0600	02447 001665 ELA,CLE,ERA	REMOVE BIT 15
0601	02450 043151R ADA XPREL	ADD COMMON LENGTH TO BASE
0602	02451 073117R STA PPREL	SET NEW RELOC ADDRESS
0603	02452 063006R LDA ID1	SAVE MAIN IDENT ADDRESS FOR
0604	02453 073072R STA IDMB5	SEGMENT REFERENCE

0605	02454	017662R	JSB DINIT	INITIALIZE DISC STORE ADDRESSES
0606	02455	016125R	JSB SPACE	
0607	02456	016715R	JSB LOAD	LOAD USER MAIN PROGRAM
0608	02457	017222R	JSB GENID	GENERATE DIRECTORY ENTRY
0609	02460	017654R	JSB REMDO	OUTPUT REMAINDER OF RECORD
0610	02461	037111R	ISZ PGCNT	
0611	02462	017672R	JSB RESET	GO RESET DSKAD TO PROPER ADDRESS
0612	02463	002400	CLA	
0613	02464	073346R	STA LIBRY	
0614	02465	063050R	LDA DSKAD	
0615	02466	073052R	STA DSKBS	
0616	02467	063135R	LDA TBREL	
0617	02470	067107R	LDB PBREL	
0618	02471	016100R	JSB BPOUT	OUTPUT BP SECTION
0619	02472	063323R	LDA TPREL	
0620	02473	073022R	STA BSPAD	SAVE PROG RELOC ADDR FOR BS
0621	02474	063135R	LDA TBREL	
0622	02475	073021R	STA BSBAD	SET BP RELOC ADDR FOR BS
0623	02476	060060	LDA .5	SET CURRENT TYPE TO 3SER SEGMENT
0624	02477	073121R	STA PTYPE	
0625	02500	063036R	LDA CURAP	
0626	02501	002004	INA	
0627	02502	073116R	STA PLMAN	SAVE CURRENT PROGRAM LIST ADDR
0628	02503	016031R	JSB INIDX	
0629	02504	016035R	JSB IDX	SET IDENT ADDRESSES
0630	02505	026556R	JMP BSEND	
0631	02506	063006R	LDA ID1	
0632	02507	073074R	STA IMAIN	SAVE MAIN BS IDENT ADDRESS
0633	02510	163013R	LDA ID6,I	
0634	02511	002021	SSA,RSS	
0635	02512	026504R	JMP BSLD	
0636	02513	010073	AND M177	
0637	02514	050060	CPA .5	SEGMENT?
0638	02515	002001	RSS	YES
0639	02516	026504R	JMP BSLD	NO- TRY NEXT ENTRY
0640	02517	163014R	LDA ID7,I	GET BS MAIN IDENT ADDRESS
0641	02520	002020	SSA	
0642	02521	003004	CMA,INA	
0643	02522	053072R	CPA IDMB8	DOES THIS CALL USER MAIN
0644	02523	002001	RSS	YES
0645	02524	026504R	JMP BSLD	NO-TRY NEXT ENTRY
0646	02525	063003R	LDA TIDNT	SAVE ADDR FOR NEXT SEGMENT SCAN
0647	02526	073015R	STA ABSID	
0648	02527	007400	CCB	
0649	02530	077070R	STB HDFLG	SET HEADING FOR SEGMENT
0650	02531	017662R	JSB DINIT	
0651	02532	016125R	JSB SPACE	
0652	02533	016726R	JSB LOADS	LOAD LUSER SEGMENT
0653	02534	017222R	JSB GENID	GENERATE DIRECTORY ENTRY
0654	02535	017654R	JSB REMDO	OUTPUT REMAINDER OF ABS RECORD
0655	02536	017672R	JSB RESET	
0656	02537	002400	CLA	
0657	02540	073346R	STA LIBRY	
0658	02541	063135R	LDA TBREL	
0659	02542	067021R	LDB BSBAD	
0660	02543	016100R	JSB BPOUT	OUTPUT BASE PAGE SECTION

0661	02544	063021R	LDA	BSPAD	
0662	02545	067135R	LDB	TBREL	
0663	02546	016036R	JSB	CLRLT	CLEAR BP LINKAGES
0664	02547	063021R	LDA	BSPAD	
0665	02550	073135R	STA	TBREL	SET BP RELOC ADDR
0666	02551	063022R	LDA	BSPAD	
0667	02552	073323R	STA	TPREL	SET PROG RELOC ADDR
0668	02553	063015R	LDA	ABSID	
0669	02554	073003R	STA	TIDNT	
0670	02555	026504R	JMP	BOLD	GO LOAD NEXT SEGMENT
0671	02556	063050R	BSEND	LDA	DSKAD
0672	02557	073051R	STA	DSKBR	SAVE CURRENT DISC ADDR OF ABS
0673	02560	063052R	LDA	DSKBS	
0674	02561	073050R	STA	DSKAD	SET CURRENT BP CODE ADDRESS
0675	02562	063021R	LDA	BSPAD	
0676	02563	067107R	LDB	PBREL	
0677	02564	016100R	JSB	BPOUT	OUTPUT BP CODE FOR MAIN
0678	02565	063051R	LDA	DSKBR	
0679	02566	073050R	STA	DSKAD	SET CURRENT ABS DISC ADDRESS
0680	02567	063107R	LDA	PBREL	
0681	02570	067021R	LDB	BSPAD	
0682	02571	016036R	JSB	CLRLT	CLEAR BP LINKAGES
0001	02572	060056	LDA	.3	
0002	02573	073121R	STA	PTYPE	SET PROG TYPE TO USER MAIN
0003	02574	016277R	JSB	CLID3	CLEAR PROG LOADED FLAG
0004	02575	063304R	LDA	PLAD	
0005	02576	073036R	STA	CURAP	RESET PLIST POINTER
0006	02577	026440R	JMP	BOLD	LOAD NEXT USER MAIN
0007	02600	016125R	BDEND	JSB	SPACE
0008	02601	063111R	LDA	PGCNT	CHECK IF ANY PROGRAMS LOADED
0009	02602	002002	SZA		SKIP=NO PROGRAMS LOADED
0010	02603	026610R	JMP	BD005	
0011	02604	063174R	LDA	.18	
0012	02605	066703R	LDB	NOLOD	
0013	02606	016101R	JSB	SYOUT	
0014	02607	026613R	JMP	BD010	
0015	02610	063172R	BD005	LDA	.14
0016	02611	066673R	LDB	CMESS	
0017	02612	016101R	JSB	SYOUT	
0018	02613	016375R	BD010	JSB	SET
0019*					
0020*	LOADER TERMINATION				
0021*					
0022	02614	002400	LTERM	CLA	
0023	02615	017421R	JSB	OUTID	
0024	02616	017537R	JSB	SETCR	SET USER DISC FLAG IN BASE PAGE
0025	02617	063053R	LDA	DSKID	
0026	02620	002003	SZA	RSS	
0027	02621	026651R	JMP	LTERZ	
0028	02622	067275R	LDB	AIBUF	
0029	02623	016243R	JSB	DISKO	OUTPUT LAST DIRECTORY BUFFER
0030	02624	063111R	LDA	PGCNT	
0031	02625	002003	SZA	RSS	
0032	02626	026651R	JMP	LTERZ	
0033	02627	063050R	LDA	DSKAD	UPDATE USER FILE T/S BOUNDARY
0034	02630	067301R	LDB	DFUTS	IN BASE PAGE

0035	02631	017543R		JSB BPSTR	
0036	02632	060155		LDA SYSSC	
0037	02633	050161		CPA CUDSC	
0038	02634	002001		RSS	
0039	02635	026641R		JMP LTERX	OK - USER NOT ON SYSTEM
0040	02636	063050R		LDA DSKAD	
0041	02637	067302R		LDB DFSTS	LIKEWISE SET SYNTS - UDNTS
0042	02640	017543R		JSB BPSTR	STORE A IN B
0043	02641	060200	LTERX	LDA DISCL	
0044	02642	067274R		LDB ADBUF	
0045	02643	016235R		JSB DISKI	
0046	02644	060157		LDA UDNTS	NEXT T/S AVAILABLE IN USER AREA
0047	02645	073426R		STA DBUF+64	
0048	02646	060200		LDA DISCL	
0049	02647	067274R		LDB ADBUF	WRITE OUT SYS BUFFER SECTOR
0050	02650	016243R		JSB DISKO	
0051	02651	060331	LTERZ	LDA SSFLG	RESTORE SS FLAG
0052	02652	053203R		CPA ASCNO	
0053	02653	026657R		JMP LTERW	
0054	02654	063127R		LDA SAVSS	
0055	02655	067204R		LDB XSSFL	
0056	02656	017543R		JSB BPSTR	RESTORE SS FLAG IN BASE PAGE
0057	02657	016001X	LTERW	JSB EXEC	REQUEST PROGRAM COMPLETION
0058	02660	002662R		DEF **2	
0059	02661	000061		DEF .6	
0060*					
0061	02662	002663R	MESAB	DEF **1	
0062	02663	046117		ASC 8,LOADR TERMINATED	
0063	02673	002674R	CMESB	DEF **1	
0064	02674	046117		ASC 7,LOADR COMPLETE	
0065	02703	002704R	NOLOD	DEF **1	
0066	02704	047117		ASC 9,NO PROGRAMS LOADED	

0068* LOAD PROVIDES ENTRY TO LOADS FOR THOSE PROGRAMS WHICH REQUIRE
 0069* USE OF A NEW BASE PAGE AND PROGRAM BASE.

0070*
 0071 02715 000000 LOAD NOP
 0072 02716 007400 CCB SET HEADING FLAG
 0073 02717 077070R STB HDFLG
 0074 02720 063117R LDA PPREL
 0075 02721 073323R STA TPREL SET CURRENT PROG RELOC ADDRESS
 0076 02722 063107R LDA PBREL
 0077 02723 073135R STA TBREL SET CURRENT BP RELOC ADDRESS
 0078 02724 016726R JSB LOADS
 0079 02725 126715R JMP LOAD,I RETURN

0080*
 0081* LOADS IS THE MAIN LOADING ROUTINE FOR GENERATING THE ABSOLUTE
 0082* CODE AND LINKING ALL CALLED SUBROUTINES. IT IS USED BY MAIN AND
 0083* SEGMENT PROGRAM TYPES. IT READS THE RELOCATABLE RECORDS FROM THE
 0084* SCRATCH PORTION OF THE DISC, AND WRITES THE ABSOLUTE CODE ON THE
 0085* USER FILE AREA OF THE DISC.

0086*
 0087 02726 000000 LOADS NOP
 0088 02727 003400 CCA
 0089 02730 073114R STA PLFLG SET NO DBL RECORDS IN FLAG
 0090 02731 002400 LOADN CLA INITIALLY SET FOR PASS 1
 0091 02732 073156R LD3 STA L001 PASS 2 WHEN LENGTH OF LOW LINK F
 0092 02733 003400 CCA SET FOR UPPER LINK
 0093 02734 073151R STA LWH1 AREA PRESENT
 0094 02735 067156R LDB L001 IF PASS 1,
 0095 02736 006002 SZB SAVE PROGRAM & BASE PAGE
 0096 02737 026744R JMP LD1 STARTING RELOCATION ADDRESSES
 0097 02740 063323R LDA TPREL
 0098 02741 073154R STA LWH4
 0099 02742 063135R LDA TBREL
 0100 02743 073153R STA LWH3
 0101 02744 063113R LD1 LDA RSS RESTORE SKIP INSTRUCTION
 0102 02745 073071R STA BLIN3
 0103 02746 002400 CLA SET FOR PASS 1 OR 2
 0104 02747 073157R LOADX STA L01 PASS 3 WHEN LOW LINK OUT TO DISC
 0105 02750 063156R LDA L001 IF PASS 2 OR 3 (IE. SOME LOADING
 0106 02751 002003 SZB,RSS HAS BEEN PERFORMED), CLEAR
 0107 02752 026756R JMP LD2 BASE PAGE LINK ADDRESSES IN
 0108 02753 063153R LDA LWH3 LST ENTRIES
 0109 02754 067135R LDB TBREL
 0110 02755 016036R JSB CLRLT
 0111 02756 063154R LD2 LDA LWH4 RESTORE ORIGINAL PROG RELOCATION
 0112 02757 067157R LDB L01 BASE, UNLESS PASS 3. IF PASS 3
 0113 02760 006003 SZB,RSS RELOCATION BASE HAS BEEN AD-
 0114 02761 073323R STA TPREL JUSTED PAST LOW LINK AREA.
 0115 02762 063153R LDA LWH3 RESTORE ORIGINAL BASE PAGE
 0116 02763 073135R STA TBREL RELOCATION BASE.
 0117 02764 016176R JSB CLIST BLANK MEMORY MAP BUFFER
 0118 02765 002400 CLA
 0119 02766 167014R LDB ID7,I
 0120 02767 006020 SSB
 0121 02770 002004 INA
 0122 02771 073137R STA TDIR
 0123 02772 063300R LDA AMLST ADDR OF MEMORY MAP BUFFER

0124	02773	073016R	STA AMAD	
0125	02774	063070R	LDA HDFLG	CHECK IF MAIN OR SEGMENT LOADING
0126	02775	073150R	STA TEMP2	
0127	02776	002021	SSA,RSS	
0128	02777	037016R	ISZ AMAD	
0129	03000	163006R	LDA ID1,I	STORE NAME INTO MEMORY MAP
0130	03001	173016R	STA AMAD,I	
0131	03002	037016R	ISZ AMAD	
0132	03003	073160R	STA ZBUF	
0133	03004	163007R	LDA ID2,I	
0134	03005	173016R	STA AMAD,I	
0135	03006	037016R	ISZ AMAD	
0136	03007	073161R	STA ZBUF+1	
0137	03010	163010R	LDA ID3,I	
0138	03011	010075	AND M1774	
0139	03012	073162R	STA ZBUF+2	
0140	03013	033210R	IOR BLANK	
0141	03014	173016R	STA AMAD,I	
0142*				
0143*	DISSET INTERROGATES BIT 15 OF WD 4 OF THE			
0144*	IDENT TABLE.			
0145*	BIT15 = 1; FLAG DISCX IS SET(USER AREA)			
0146*	BIT15 = 0; FLAG DISCX IS CLEARED(SYSTEM AREA)			
0147*				
0148	03015	064056	LDB .3	
0149	03016	163011R	LDA ID4,I	
0150	03017	002021	SSA,RSS	
0151	03020	064055	LDB .2	
0152	03021	077046R	STB DISCX	
0153*				
0154	03022	063157R	LDA L01	PASS 3?
0155	03023	002002	SZA	
0156	03024	027030R	JMP LD4	YES - TEST REAL FLAG
0157	03025	037150R	ISZ TEMP2	NO - TEST TEMPORARY HDFLG
0158	03026	027072R	JMP SUBHD	
0159	03027	027032R	JMP LD5	
0160	03030	037070R	LD4 ISZ HDFLG	TEST REAL HDFLG
0161	03031	027072R	JMP SUBHD	
0162*				
0163	03032	063050R	LD5 LDA DSKAD	GET CURRENT DISC ADDRESS
0164	03033	067157R	LDB L01	IF NOT PASS 3,
0165	03034	006003	SZB,RSS	
0166	03035	073054R	STA DSKMN	SAVE
0167	03036	063121R	LDA PTYPE	CHECK PROGRAM TYPE
0168	03037	050060	CPA .5	
0169	03040	027045R	JMP BGSPL	SEGMENT SO CLEAR PLIST
0170	03041	063272R	LDA N128	
0171	03042	073113R	STA PLCNT	SET PLIST WORD COUNT
0172	03043	067304R	LDB PLAD	
0173	03044	027054R	JMP CPLST	CLEAR PROGRAM LIST
0174	03045	063304R	BGSPL LDA PLAD	
0175	03046	003004	CMA,INA	
0176	03047	043116R	ADA PLMAN	
0177	03050	043272R	ADA N128	
0178	03051	073113R	STA PLCNT	
0179	03052	067116R	LDB PLMAN	

0180	03053	027062R		JMP CPXXX	
0181	03054	063006R	CPLST	LDA ID1	GET IDENT ADDR OF MAIN AND SET
0182	03055	170001		STA B,I	INTO PLIST
0183	03056	006004		INB	INCR CURRENT PLIST ADDRESS
0184	03057	037113R		ISZ PLCNT	SKIP-PLIST FULL
0185	03060	002001		RSS	
0186	03061	027067R		JMP SETMP	NO PLIST CLEARING REQUIRED
0187	03062	002400	CPXXX	CLA	
0188	03063	170001		STA B,I	CLEAR PLIST WORD
0189	03064	006004		INB	
0190	03065	037113R		ISZ PLCNT	
0191	03066	027063R		JMP *-3	
0192	03067	063207R	SETMP	LDA BLNKS	
0193	03070	072203R		STA MLIST+3	
0194	03071	072204R		STA MLIST+4	
0195	03072	063154R	SUBHD	LDA LWH4	
0196	03073	067277R		LDB AMEM5	
0197	03074	016130R		JSB CONVD	CONVERT LOAD ADDRESS TO ASCII
0198*					
0199	03076	063157R		LDA L01	PASS 3?
0200	03076	002003		SZA,RSS	
0201	03077	027103R		JMP LD6	NO- SKIP MEMORY MAP
0202	03100	063173R		LDA .16	
0203	03101	067300R		LDB AMLST	
0204	03102	016134R		JSB DRKEY	OUTPUT LINE TO LIST UNIT
0205	03103	163012R	LD6	LDA ID5,I	
0206	03104	016471R		JSB INNAM	INPUT NAM RECORD
0207	03105	063035R		LDA CURAL	GET RECORD TYPE
0208	03106	002004		INA	
0209	03107	160000		LDA A,I	
0210	03110	001723		ALF,RAR	
0211	03111	010062		AND .7	
0212	03112	050054		CPA .1	NAME RECORD?
0213	03113	002001	RSS	RSS	
0214	03114	026641R		JMP NMERR	
0215	03115	063035R		LDA CURAL	
0216	03116	040061		ADA .6	
0217	03117	160000		LDA A,I	
0218	03120	073115R		STA PLGTH	SAVE PROGRAM LENGTH
0219	03121	016724R		JSB ALINK	SET UP LOW LINK AREA FOR CP LINK
0220	03122	067323R		LDB TPREL	
0221	03123	077123R		STB RELAD	
0222	03124	063346R		LDA LIBRY	SET CURRENT RELOCATION ADDRESS
0223	03125	002002		SZA	
0224	03126	027142R		JMP BPN	
0225	03127	063035R		LDA CURAL	
0226	03130	040064		ADA .9	
0227	03131	160000		LDA A,I	
0228	03132	010073		AND M177	
0229	03133	050061		CPA .6	
0230	03134	027140R		JMP SUBH5	
0231	03135	050062		CPA .7	
0232	03136	002001		RSS	
0233	03137	027142R		JMP BPN	
0234	03140	063323R	SUBH5	LDA TPREL	
0235	03141	073346R		STA LIBRY	

0236	03142	163035R	BPN	LDA CURAL,I	UPDATE TO END OF NAM RECORED
0237	03143	001727		ALF,ALF	
0238	03144	010074		AND M377	
0239	03145	070001		STA B	
0240	03146	063035R		LDA CURAL	
0241	03147	040001		ADA B	
0242	03150	073035R		STA CURAL	SET TO END OF NAM RECORD
0243	03151	063077R		LDA LCNT	
0244	03152	040001		ADA B	
0245	03153	073077R		STA LCNT	SET NEW CURRENT COUNT
0246	03154	016463R	CLSRC	JSB DBSET	
0247	03155	163035R		LDA CURAL,I	
0248	03156	064000		LDB A	
0249	03157	001723		ALF,RAR	
0250	03160	010062		AND .7	
0251	03161	050055		CPA .2	
0252	03162	027354R		JMP DENTR	GO TO ENT RECORD PROCESSOR
0253	03163	050056		CPA .3	
0254	03164	027550R		JMP DDBLR	GO TO DBL RECORD PROCESSOR
0255	03165	050057		CPA .4	
0256	03166	027355R		JMP DEXTR	GO TO EXT RECORD PROCESSOR
0257	03167	050060		CPA .5	
0258	03170	027172R		JMP DENDR	GO TO END RECORD PROCESSOR
0259	03171	026641R		JMP NMERR	
0260	03172	063157R	DENDR	LDA L01	IF END DETECTED DURING PASS 2,
0261	03173	002002		SZA	SET LWH1=0 TO INDICATE NO
0262	03174	027177R		JMP LD7	UPPER LINK AREA, THEN OUTPUT
0263	03175	073151R		STA LWH1	LOW LINK AREA TO
0264	03176	027056R		JMP CROSS	DISC
0265	03177	163035R	LD7	LDA CURAL,I	
0266	03200	010064		AND .1	
0267	03201	073223R		STA PENTZ	
0268	03202	016463R		JSB DBSET	
0269	03203	016463R		JSB DBSET	
0270	03204	037123R		ISZ RELAD	
0271	03205	163035R		LDA CURAL,I	GET TRANSFER ADDRESS
0272	03206	043323R		ADA TPREL	
0273	03207	067070R		LDB HDFLG	
0274	03210	006002		SZB	SKIP- IF MAIN OR SEGMENT
0275	03211	027225R		JMP DEN20	
0276	03212	067223R		LDB PENTZ	
0277	03213	006002		SZB	
0278	03214	027224R		JMP DEN10	YES
0279	03215	063154R		LDA UMFWA	SET PRIMARY ENTRY FOR DRIVER
0280	03216	067156R		LDB DVR	IS THIS TYPE 4 -
0281	03217	054057		CPB .4	DEVICE DRIVER?
0282	03220	027224R		JMP DEN10	YES - IGNORE L08 ERROR
0283	03221	063172R		LDA ERR08	NO SO OUTPUT ERROR MESSAGE
0284	03222	026370R		JMP ABRTX	ABORT LOADR
0285	03223	000000	PENTZ	NOP	PRIMARY ENTRY PRESENT FLAG
0286	03224	073120R	DEN10	STA PRENT	SAVE PRIMARY ENTRY POINT
0287	03225	017721R	DEN20	JSB DEBUG	
0288	03226	016000R		JSB INLST	
0289	03227	016004R	CLST	JSB LSTX	SET LST ADDRESS
0290	03230	027235R		JMP LSTCR	END OF LST
0291	03231	162777R		LDA LST3,I	

0292	03232	010075		AND M1774	CLEAR LST ORDINAL
0293	03233	172777R		STA LST3,I	
0294	03234	027227R		JMP CLST	CONTINUE CLEARING ORDINAL
0295	03235	063115R	LSTCR	LDA PLGTH	GET PROGRAM LENGTH
0296	03236	001265		RAL,CLE,ERA	SET E=SIGN
0297	03237	043323R		ADA TPREL	ADD PROGRAM RELOCATION BASE
0298	03240	002040		SEZ	SKIP IF ASSEMBLER PRODUCED
0299	03241	063123R		LDA RELAD	GET CURRENT RELOCATION ADDRESS
0300	03242	073323R		STA TPREL	SET NEW RELOCATION BASE
0301	03243	070001		STA B	
0302	03244	063123R		LDA RELAD	GET LAST WORD ADDRESS
0303	03245	003004		CMA,INA	
0304	03246	044000		ADB A	ADD RELOCATION BASE
0305	03247	006003		SZB,RSS	
0306	03250	027254R		JMP **4	
0307	03251	002400		CLA	
0308	03252	067323R		LDB TPREL	
0309	03253	017445R		JSB LABD0	
0310	03254	063157R		LDA L01	IF PASS 3,
0311	03255	002021		SSA,RSS	OUTPUT UPPER
0312	03256	017115R		JSB BLINK	LINK AREA
0313	03257	063272R		LDA N128	
0314	03260	073113R		STA PLCNT	SET PROG LIST COUNTER
0315	03261	067304R		LDB PLAD	GET ADDR OF PROGRAM LIST
0316	03262	160001	PLSCN	LDA B,I	GET PROGRAM LIST ENTRY
0317	03263	002003		SZA,RSS	SKIP=ENTRY IN PLIST
0318	03264	027277R		JMP DEIDG	
0319	03265	002020		SSA	SKIP IF PROG LOADED
0320	03266	027336R		JMP ENTID	GET NEXT PROG IDENT ADDRESS
0321	03267	006004	EPLST	INB	INCR PROGRAM LIST ADDRESS
0322	03270	037113R		ISZ PLCNT	SKIP IF END OF PROGRAM LIST
0323	03271	027262R		JMP PLSCN	NO=CONTINUE PROG LIST SEARCH
0324	03272	063121R	CLFLG	LDA PTYPE	GET CURRENT PROGRAM TYPE
0325	03273	050056		CPA .3	USER MAIN
0326	03274	126726R		JMP LOADS,I	RETURN
0327	03275	016277R		JSB CLID3	CLEAR PROG LOADED FLAGS
0328	03276	126726R		JMP LOADS,I	RETURN
0329	03277	063041R	DEIDG	LDA DBFLG	CHECK IF DEBUG TO BE LOADED
0330	03300	002003		SZA,RSS	
0331	03301	027272R		JMP CLFLG	NO
0332	03302	063030R		LDA CLXXX	CHECK IF ALREADY LOADED
0333	03303	002002		SZA	
0334	03304	027272R		JMP CLFLG	YES
0335	03305	037030R		ISZ CLXXX	INCR DEBUG LOADED FLAG
0336	03306	077140R		STB TEMP	
0337	03307	016000R		JSB INLST	SEARCH FOR DEBUG IN LST
0338	03310	016004R	DEI10	JSB LSTX	
0339	03311	026637R		JMP RCERR	
0340	03312	162775R		LDA LST1,I	
0341	03313	053217R		CPA CHRDE	
0342	03314	002001		RSS	
0343	03315	027310R		JMP DEI10	TRY NEXT LST
0344	03316	162776R		LDA LST2,I	
0345	03317	053220R		CPA CHRDE+1	
0346	03320	002001		RSS	
0347	03321	027310R		JMP DEI10	TRY NEXT LST

0348	03322	162777R	LDA LST3,I	
0349	03323	010075	AND M1774	
0350	03324	070001	STA B	
0351	03325	063221R	LDA CHRDE+2	
0352	03326	010075	AND M1774	
0353	03327	050001	CPA B	
0354	03330	002001	RSS	DEBUG FOUND
0355	03331	027310R	JMP DEI10	TRY NEXT ENTRY
0356	03332	067140R	LDB TEMP	
0357	03333	163000R	LDA LST4,I	
0358	03334	033205R	IOR MSIGN	
0359	03335	170001	STA B,I	SET IDENT ADDRESS IN PLIST TABLE
0360	03336	001265	ENTID RAL,CLE,ERA	CLEAR SIGN(PROG TO BE LOADED)
0361	03337	073003R	STA TIDNT	SET IDENT ADDR FOR IDX
0362	03340	072210R	STA TBUF	SAVE CURRENT IDENT ADDRESS
0363	03341	077036R	STB CURAP	SAVE CURRENT PLIST ADDRESS
0364	03342	016035R	JSB IDX	SET IDENT ADDRESSES
0365	03343	026637R	JMP RCERR	
0366	03344	067036R	LDB CURAP	GET CURRENT PLIST ADDRESS
0367	03345	163010R	LDA ID3,I	GET PROGRAM USAGE FLAG
0368	03346	002014	SLA,INA	SKIP=PROG NOT YET LOADED
0369	03347	027267R	JMP EPLST	TEST FOR END OF PLIST
0370	03350	173010R	STA ID3,I	SET PROG USAGE FLAG=LOADED
0371	03351	062210R	LDA TBUF	GET CURRENT IDENT ADDRESS
0372	03352	170001	STA B,I	SET ENTRY IN PROG LIST
0373	03353	026731R	JMP LOADN	LOAD NEXT PROGRAM
0374	03354	003401	DENTR CCA,RSS	SET ENT FLAG
0375	03355	002400	DEXTR CLA	SET EXT FLAG
0376	03356	073104R	STA NXFLG	SAVE ENT/EXT FLAG
0377	03357	060001	LDA B	
0378	03360	010071	AND M37	
0379	03361	003004	CMA,INA	
0380	03362	073065R	STA EXCNT	SET SYMBOL COUNTER
0381	03363	016463R	JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0382	03364	016463R	JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0383	03365	163035R	NXSYM LDA CURAL,I	
0384	03366	072210R	STA TBUF	SAVE NAME 1,2
0385	03367	016463R	JSB DBSET	
0386	03370	163035R	LDA CURAL,I	
0387	03371	072211R	STA TBUF+1	SAVE NAME 3,4
0388	03372	016463R	JSB DBSET	
0389	03373	163035R	LDA CURAL,I	
0390	03374	072212R	STA TBUF+2	SAVE NAME 5
0391	03375	016463R	JSB DBSET	
0392	03376	016000R	JSB INLST	INITIALIZE LSTX
0393	03377	016004R	ENEX1 JSB LSTX	SET LST ADDRESSES
0394	03400	027421R	JMP XXXX1	
0395	03401	062210R	LDA TBUF	
0396	03402	152775R	CPA LST1,I	NAME 1,2 EQUAL?
0397	03403	002001	RSS	YES
0398	03404	027377R	JMP ENEX1	NO-TRY NEXT LST ENTRY
0399	03405	062211R	LDA TBUF+1	
0400	03406	152776R	CPA LST2,I	NAME 3,4 EQUAL?
0401	03407	002001	RSS	YES
0402	03410	027377R	JMP ENEX1	NO-TRY NEXT LST ENTRY
0403	03411	062212R	LDA TBUF+2	

0404	03412	010075	AND M1774	
0405	03413	070001	STA B	
0406	03414	162777R	LDA LST3,I	UPPER CHAR OF
0407	03415	010075	AND M1774	NAME 5 EQUAL?
0408	03416	050001	CPA B	
0409	03417	002001	RSS	YES
0410	03420	027377R	JMP ENEX1	NO-TRY NEXT LST ENTRY
0411	03421	063104R	XXXX1 LDA NXFLG	
0412	03422	002003	SZA,RSS	SKIP IF ENT
0413	03423	027462R	JMP EXT2	GO TO PROCESS EXT
0414	03424	067323R	LDB TPREL	GET CURRENT PROG RELOC BASE
0415	03425	147035R	ADB CURAL,I	ADD CURRENT RELOC VALUE
0416	03426	077105R	STB OPRND	SAVE ABS ENT PT ADDRESS
0417	03427	163001R	LDA LST5,I	GET BP LINK ADDRESS
0418	03430	002003	SZA,RSS	SKIP-BP LINK ADDR SET
0419	03431	016343R	JSB BPSCN	SET BP LINK ADDR,BP LINK
0420	03432	173001R	STA LST5,I	SET BP LINK ADDR IN LST
0421	03433	016433R	JSB SETOP	SET OPERAND IN BASE PAGE
0422	03434	063157R	LDA L01	UNLESS IT'S
0423	03435	002003	SZA,RSS	PASS 3, SKIP
0424	03436	027460R	JMP NLENT	ENT LIST
0425	03437	016176R	JSB CLIST	CLEAR MEMORY MAP BUFFER
0426	03440	063206R	LDA BLAST	
0427	03441	072201R	STA MLIST+1	
0428	03442	162775R	LDA LST1,I	SET NAME 1,2 IN MEMORY MAP
0429	03443	072202R	STA MLIST+2	
0430	03444	162776R	LDA LST2,I	SET NAME 3,4 IN MEMORY MAP
0431	03445	072203R	STA MLIST+3	
0432	03446	162777R	LDA LST3,I	
0433	03447	010075	AND M1774	
0434	03450	033210R	IOR BLANK	
0435	03451	072204R	STA MLIST+4	SET NAME 5 IN MEMORY MAP
0436	03452	063105R	LDA OPRND	GET ABSOLUTE ENTRY PT ADDRESS
0437	03453	067277R	LDB AMEM5	
0438	03454	016130R	JSB CONVD	CONVERT TO ASCII
0439	03455	063173R	LDA .16	
0440	03456	067300R	LDB AMLST	
0441	03457	016134R	JSB DRKEY	PRINT ENTRY POINT
0442	03460	016463R	NLENT JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0443	03461	027545R	JMP EXEND	GO PROCESS NEXT SYMBOL
0444	03462	062212R	EXT2 LDA TBUF+2	GET ORDINAL
0445	03463	172777R	STA LST3,I	SET ORDINAL IN LST
0446	03464	163001R	LDA LST5,I	GET BP LINK ADDRESS
0447	03465	002003	SZA,RSS	SKIP-BP LINK ADDRESS SET
0448	03466	016444R	JSB ALLOC	ALLOCATE NEW BP LOCATION
0449	03467	173001R	STA LST5,I	SET NEW BP LINK ADDRESS
0450	03470	163000R	LDA LST4,I	GET IDENT ADDRESS
0451	03471	002002	SZA	
0452	03472	002020	SSA	
0453	03473	027545R	JMP EXEND	IGNORE UNDEFINED PROGRAM CALL
0454	03474	073003R	STA TIDNT	SET ID ADDR FOR IDX
0455	03475	063006R	LDA ID1	GET CURRENT IDENT ADDR
0456	03476	072210R	STA TBUF	SAVE CURRENT IDENT ADDRESS
0457	03477	016035R	JSB IDX	
0458	03500	027545R	JMP EXEND	
0459	03501	163013R	LDA ID6,I	GET M/S, TYPE

0460	03502	072211R	STA TBUF+1	
0461	03503	163010R	LDA ID3,I	GET PROGRAM USAGE FLAG
0462	03504	072212R	STA TBUF+2	
0463	03505	062210R	LDA TBUF	GET CURRENT IDENT ADDRESS
0464	03506	073003R	STA TIDNT	
0465	03507	016035R	JSB IDX	
0466	03510	027545R	JMP EXEND	
0467	03511	062211R	LDA TBUF+1	GET M/S, TYPE FOR EXT
0468	03512	001265	RAL,CLE,ERA	SET E=M/S
0469	03513	010073	AND M177	ISOLATE TYPE
0470	03514	050061	CPA .6	
0471	03515	027522R	JMP LIBUT	LIBRARY-TEST FOR LOADING
0472	03516	050062	CPA .7	
0473	03517	027522R	JMP LIBUT	
0474	03520	002040	SEZ	SKIP-NOT MAIN PROGRAM
0475	03521	027545R	JMP EXEND	IGNORE PROGRAM CALL
0476	03522	063272R	LIBUT LDA N128	
0477	03523	073113R	STA PLCNT	SET PROGRAM LIST COUNTER
0478	03524	067304R	LDB PLAD	GET PROGRAM LIST ADDRESS
0479	03525	160001	PLEXT LDA B,I	GET PROG LIST ENTRY
0480	03526	002003	SZA,RSS	SKIP IF IDENT ADDRESS
0481	03527	027537R	JMP SEXT	MAKE ENTRY IN PROG LIST
0482	03530	001265	RAL,CLE,ERA	CLEAR SIGN
0483	03531	153000R	CPA LST4,I	PROG PREVIOUSLY CALLED?
0484	03532	027545R	JMP EXEND	YES=IGNORE EXT SYMBOL
0485	03533	006004	INB	INCR PROG LIST ADDR
0486	03534	037113R	ISZ PLCNT	END OF PROG LIST?
0487	03535	027525R	JMP PLEXT	NO=GO GET NEXT PROGRAM
0488	03536	027545R	JMP EXEND	TEST FOR ALL EXT PROCESSED
0489	03537	062212R	SEXT LDA TBUF+2	GET PROGRAM USAGE FLAG
0490	03540	000010	SLA	SKIP=PROGRAM NOT LOADED
0491	03541	027545R	JMP EXEND	OMIT PROGRAM LIST ENTRY
0492	03542	163000R	LDA LST4,I	GET CALLED PROG IDENT ADDR
0493	03543	033205R	IOR MSIGN	SET SIGN TO SHOW NOT LOADED
0494	03544	170001	STA B,I	SET IN PROGRAM LIST
0495	03545	037065R	EXEND ISZ EXCNT	SKIP=ALL SYMBOLS PROCESSED
0496	03546	027365R	JMP NXSYM	NO=PROCESS NEXT SYMBOL
0497	03547	027154R	JMP CLSRC	NO=CLASSIFY NEXT RECORD
0498	03550	060001	DDBLR LDA B	
0499	03551	010072	AND M77	
0500	03552	003004	CMA,INA	
0501	03553	073065R	STA EXCNT	SET INSTRUCTION COUNT
0502	03554	016463R	JSB DBSET	
0503	03555	016463R	JSB DBSET	
0504	03556	167035R	LDB CURAL,I	GET RELOCATION ADDRESS
0505	03557	077042R	STB DBLAD	SAVE RELOCATION ADDRESS
0506	03560	037114R	ISZ PLFLG	SKIP-FIRST DBL RECORD
0507	03561	027574R	JMP DBL0	IGNORE SUBSEQUENT RECORDS
0508	03562	063157R	LDA L01	
0509	03563	002002	SZA	
0510	03564	163013R	LDA ID6,I	GET TYPE
0511	03565	010073	AND M177	
0512	03566	050060	CPA .5	
0513	03567	002001	RSS	SET NEW MEMORY BOUND FOR SEGMENT
0514	03570	006400	CLB	SET PGMA0=0
0515	03571	077023R	HABSS STB BSSDP	SAVE INITIAL PROG DISPLACEMENT

0516	03572	047154R		ADB LWH4	
0517	03573	077027R		STB CBASE	SAVE CURRENT BASE PROGRAM ADDRESS
0518	03574	016463R	DBL0	JSB DBSET	
0519	03575	167035R	DBL1	LDB CURAL,I	GET RELOCATION BYTES
0520	03576	077124R		STB REKEY	SAVE FOR RELOCATION TYPE
0521	03577	060046		LDA N5	
0522	03600	073075R		STA INSCN	SET RELOCATION BYTE COUNT
0523	03601	016463R		JSB DBSET	
0524	03602	063124R	DBL2	LDA REKEY	GET RELOCATION BYTES
0525	03603	001723		ALF,RAR	
0526	03604	073124R		STA REKEY	SAVE FOR NEXT INST WORD
0527	03605	010062		AND .7	ISOLATE CURRENT BYTE
0528	03606	050057		CPA .4	EXTERNAL REFERENCE?
0529	03607	027635R		JMP DBL4	YES-GET LINK ADDRESS
0530	03610	050060		CPA .5	MEMORY REFERENCE?
0531	03611	027656R		JMP DBL6	YES-CHECK FOR INDIRECT LINK
0532	03612	043321R		ADA RBTAD	
0533	03613	160000		LDA A,I	GET RELOCATION BASE
0534	03614	143035R		ADA CURAL,I	ADD CURRENT INSTRUCTION WORD
0535	03615	067042R	DBL3	LDB DBLAD	CURRENT RELOCATION BASE
0536	03616	047323R		ADB TPREL	ADD CURRENT PROG RELOCATION BASE
0537	03617	073150R		STA TEMP2	SAVE A-REGISTER
0538	03620	063157R		LDA L01	UNLESS IT'S
0539	03621	002003		SZA,RSS	PASS 3, DON'T
0540	03622	027625R		JMP LD8	OUTPUT TO DISC
0541	03623	063150R		LDA TEMP2	RESTORE A-REGISTER
0542	03624	017445R		JSB LABD0	OUTPUT ABSOLUTE WORD
0543	03625	016463R	LDB	JSB DBSET	
0544	03626	037065R		ISZ EXCNT	SKIP-LAST INSTRUCTION OUT
0545	03627	002001		RSS	NO-CONTINUE
0546	03630	027154R		JMP CLSRC	YES-CLASSIFY NEXT RECORD
0547	03631	037042R		ISZ DBLAD	INCR DBL RELOCATION ADDRESS
0548	03632	037075R		ISZ INSCN	SKIP IF NEW RELOCATION BYTE
0549	03633	027602R		JMP DBL2	PROCESS NEXT INSTRUCTION
0550	03634	027575R		JMP DBL1	GET NEXT RELOCATION BYTE
0551	03635	163035R	DBL4	LDA CURAL,I	GET CURRENT DBL WORD
0552	03636	010074		AND M377	
0553	03637	073066R		STA EXORD	ISOLATE AND SAVE ORDINAL
0554	03640	016000R		JSB INLST	INITIALIZE LSTX
0555	03641	016004R	DBL5	JSB LSTX	SET LST ADDRESSES
0556	03642	027615R		JMP DBL3	ORDINAL NOT FOUND
0557	03643	162777R		LDA LST3,I	GET WORD3 OF LST ENTRY
0558	03644	010074		AND M377	
0559	03645	053066R		CPA EXORD	SAME ORDINAL?
0560	03646	002001		RSS	YES
0561	03647	027641R		JMP DBL5	NO-CONTINUE SEARCH
0562	03650	167001R		LDB LST5,I	GET BP LINK ADDRESS
0563	03651	163035R		LDA CURAL,I	GET CURRENT DBL WORD
0564	03652	013204R		AND M1740	ISOLATE INSTRUCTION CODE
0565	03653	033205R		IOR MSIGN	SET INDIRECT
0566	03654	030001		IOR B	SET ADDRESS=BP LINK ADDRESS
0567	03655	027615R		JMP DBL3	OUTPUT INSTRUCTION
0568	03656	163035R	DBL6	LDA CURAL,I	GET CURRENT DBL WORD
0569	03657	073076R		STA INSTR	SAVE INSTRUCTION CODE
0570	03660	016463R		JSB DBSET	GET ADDR OF NEXT WORD IN LBUF
0571	03661	063076R		LDA INSTR	GET INSTRUCTION CODE

0572	03662	067323R	LDB TPREL	GET CURRENT PROG RELOC ADDR
0573	03663	001332	RAR,SLA,RAL	SKIP IF PROGRAM REFERENCE
0574	03664	067325R	LDB COMAD	REPLACE PROG WITH COMMON BASE
0575	03665	147035R	ADB CURAL,I	ADD CURRENT INSTR ADDRESS
0576	03666	013204R	AND M1740	
0577	03667	002020	SSA	SKIP IF DIRECT REFERENCE
0578	03670	047205R	ADB MSIGN	SET INDIRECT BIT
0579	03671	073076R	STA INSTR	SAVE INSTRUCTION CODE
0580	03672	063323R	LDA TPREL	
0581	03673	043042R	ADA DBLAD	ISOLATE PAGE #
0582	03674	013203R	AND M0760	SAVE CURRENT PAGE #
0583	03675	073106R	STA PAGNO	GET INSTRUCTION ADDRESS
0584	03676	060001	LDA B	ISOLATE PAGE #
0585	03677	013203R	AND M0760	SAME PAGE?
0586	03700	053106R	CPA PAGNO	YES=NO INDIRECT ADDRESS NEEDED
0587	03701	027714R	JMP DBL7	SAVE INSTRUCTION ADDRESS
0588	03702	077105R	STB OPRND	SCAN BASE PAGE FOR SAME OPERAND
0589	03703	016343R	JSB BPSCN	SET OPERAND IN BP LOCATION
0590	03704	016433R	LINKC JSB SETOP	SAVE LINK ADDRESS
0591	03705	070001	STA B	GET PAGE OFFSET
0592	03706	013201R	AND 01777	CURRENT PAGE?
0593	03707	050001	CPA B	NO= SKIP
0594	03710	002001	RSS	SET Z/C = 1 (CURRENT)
0595	03711	033202R	IOR M2000	SET INDIRECT BIT
0596	03712	033205R	IOR MSIGN	GO ADD INSTRUCTION & STORE WORD
0597	03713	027717R	JMP DBL8	GET OPERAND ADDRESS
0598	03714	060001	DBL7 LDA B	
0599	03715	013201R	AND 01777	
0600	03716	033202R	IOR M2000	SET Z/C=1 (CURRENT)
0601	03717	033076R	DBL8 IOR INSTR	ADD INSTRUCTION CODE
0602	03720	027615R	JMP DBL3	OUTPUT INSTRUCTION WORD

```

0604*      DEBUG CHECKS IF CURRENT PROGRAM LOADED WAS DEBUG.
0605*      THE PRIMARY ENTRY POINT OF DEBUG IS SET INTO PRENT, AND THE
0606*      ADDRESS OF DEBUG IS SET TO BE INDIRECT.
0607*      THE PROGRAM PRIMARY ENTRY POINT IS SET INTO DEBUF
0608      03721 000000  DEBUG NOP
0609      03722 063160R  LDA ZBUF          CHECK IF
0610      03723 053217R  CPA CHRDE          -DEBUG- PROGRAM
0611      03724 002001  RSS
0612      03725 127721R  JMP DEBUG,I    NO-RETURN
0613      03726 063161R  LDA ZBUF+1
0614      03727 053220R  CPA CHRDE+1
0615      03730 002001  RSS
0616      03731 127721R  JMP DEBUG,I    NO-RETURN
0001      03732 063162R  LDA ZBUF+2
0002      03733 053221R  CPA CHRDE+2
0003      03734 002001  RSS
0004      03735 127721R  JMP DEBUG,I    NO-RETURN
0005      03736 016000R  JSB INLST
0006      03737 016004R  DSCAN JSB LSTX      FIND -DEBUG- ENTRY IN LST FOR
0007      03740 026637R  JMP RCERR
0008      03741 162775R  LDA LST1,I
0009      03742 053217R  CPA CHRDE
0010      03743 002001  RSS
0011      03744 027737R  JMP DSCAN
0012      03745 162776R  LDA LST2,I
0013      03746 053220R  CPA CHRDE+1
0014      03747 002001  RSS
0015      03750 027737R  JMP DSCAN
0016      03751 162777R  LDA LST3,I
0017      03752 010078  AND M1774
0018      03753 053221R  CPA CHRDE+2
0019      03754 002001  RSS
0020      03755 027737R  JMP DSCAN
0021      03756 037043R  ISZ DBUGF
0022      03757 163000R  LDA LST4,I    GET ENT FOR DEBUG
0023      03760 002020  SSA
0024      03761 026637R  JMP RCERR
0025      03762 167001R  LDB LST5,I    GET BP LINK ADDR OF -DEBUG-
0026      03763 047026R  ADB BPOIT
0027      03764 047205R  ADB BPBUF
0028      03765 077140R  STB TEMP      SAVE BASE PAGE LINK ADDRESS
0029      03766 164001  LDB B,I
0030      03767 063120R  LDA PRENT
0031      03770 017445R  JSB LABDO
0032      03771 163140R  LDA TEMP,I
0033      03772 033205R  IOR MSIGN
0034      03773 173140R  STA TEMP,I
0035      03774 163035R  LDA CURAL,I
0036      03775 043323R  ADA TPREL
0037      03776 073120R  STA PRENT
0038      03777 127721R  JMP DEBUG,I

```

```

0040*      INLST SETS THE ADDRESS OF THE FIRST ENTRY IN LST IN TLST
0041*      CALLING SEQUENCE
0042*      JSB INLST
0043  04000 000000  INLST NOP
0044  04001 062772R      LDA BLST
0045  04002 072773R      STA TLST
0046  04003 126000R      JMP INLST,I
0047*
0048*      LSTX SETS THE CURRENT LST ADDR INTO LST 1-5
0049*      CALLING SEQUENCE
0050*      JSB LSTX
0051*      RETURN
0052*      (P+1) IF AT END OF LST
0053*      (P+2) IF NOT AT END OF LST
0054*
0055  04004 000000  LSTX  NOP
0056  04005 062773R      LDA TLST      CHECK IF AT END OF LOADER SYMBOL
0057  04006 052774R      CPA PLST      TABLE LIST
0058  04007 002001      RSS
0059  04010 036004R      ISZ LSTX      YES, SO INCR TO DONE EXIT
0060  04011 072775R      STA LST1      SET LOADER SYMBOL TABLE ADDRESS
0061  04012 002004      INA
0062  04013 072776R      STA LST2
0063  04014 002004      INA
0064  04015 072777R      STA LST3
0065  04016 002004      INA
0066  04017 073000R      STA LST4
0067  04020 002004      INA
0068  04021 073001R      STA LST5
0069  04022 002004      INA
0070  04023 072773R      STA TLST
0071  04024 003004      CMA,INA      CHECK IF IDENT AND LST TABLES
0072  04025 043004R      ADA PIDNT      OVERLAP
0073  04026 002021      SSA,RSS
0074  04027 126004R      JMP LSTX,I      RETURN
0075  04030 026367R      JMP LOVER      SYMBOL TABLE OVERFLOW
0076*
0077*      INIDX SETS THE ADDRESS OF THE FIRST ENTRY IN THE PROGRAM
0078*      IDENT TABLE AS THE CURRENT ADDRESS.
0079*      CALLING SEQUENCE
0080*      JSB INIDX
0081  04031 000000  INIDX NOP
0082  04032 063002R      LDA BIDNT
0083  04033 073003R      STA TIDNT
0084  04034 126031R      JMP INIDX,I
0085*      IDX SETS THE ADDRESSES OF THE CURRENT 6 WORD ENTRY IN THE
0086*      IDENT TABLE FROM THE ADDRESS OF THE CURRENT ENTRY
0087*      CALLING SEQUENCE :
0088*      JSB IDX
0089*      RETURN:
0090*      (P+1) IF AT END OF IDENT TABLE
0091*      (P+2) IF NOT AT END OF IDENT TABLE
0092  04035 000000  IDX  NOP
0093  04036 063003R      LDA TIDNT
0094  04037 053004R      CPA PIDNT
0095  04040 002001      RSS

```

0096	04041	036035R	ISZ IDX
0097	04042	073006R	STA ID1
0098	04043	002004	INA
0099	04044	073007R	STA ID2
0100	04045	002004	INA
0101	04046	073010R	STA ID3
0102	04047	002004	INA
0103	04050	073011R	STA ID4
0104	04051	002004	INA
0105	04052	073012R	STA ID5
0106	04053	002004	INA
0107	04054	073013R	STA ID6
0108	04055	002004	INA
0109	04056	073014R	STA ID7
0110	04057	063003R	LDA TIDNT
0111	04060	067003R	LDB TIDNT
0112	04061	003004	CMA,INA
0113	04062	042774R	ADA PLST
0114	04063	002021	SSA,RSS
0115	04064	026070R	JMP **4
0116	04065	044044	ADB N7
0117	04066	077003R	STB TIDNT
0118	04067	126035R	JMP IDX,I
0119	04070	063100R	LDA LDING
0120	04071	002002	SZA
0121	04072	026076R	JMP **4
0122	04073	063175R	LDA ERR11
0123	04074	016114R	JSB ERROR
0124	04075	026371R	JMP ABORT
0125	04076	003400	CCA
0126	04077	042035R	ADA IDX
0127	04100	124000	JMP A,I

CHECK=PROGRAM LOADING FLAG

LOADING SO RETURN TO P+1
FOR PROGRAM PROCESSING

0129* SYOUT PRINTS MESSAGE ON SYSTEM TELETYPE. THESE INCLUDE ALL
 0130* ERROR DIAGNOSTICS AND OPERATOR ACTION MESSAGES.

0131* CALL SEQUENCE:

0132* LDA <# CHARACTERS IN MESSAGE (+)>

0133* LDB <MESSAGE ADDRESS>

0134* JSB SYOUT

0135 04101 000000 SYOUT NOP

0136 04102 076111R STB SYM

0137 04103 003004 CMA,INA

0138 04104 073143R STA TTYNO

0139 04105 016001X JSB EXEC

0140 04106 004113R DEF **5

0141 04107 000055 DEF .2

0142 04110 000054 DEF .1

0143 04111 004111R SYM DEF *

0144 04112 011143R DEF TTYNO

0145 04113 126101R JMP SYOUT,I RETURN

0146* ERROR OUTPUTS ERROR CODE TO SYSTEM TELETYPE

0147* CALLING SEQUENCE:

0148* LDA <2 DIGIT ASCII ERROR CODE>

0149* JSB ERROR

0150 04114 000000 ERROR NOP

0151 04115 072124R STA MESSE+2

0152 04116 060057 LDA .4

0153 04117 066122R LDB MESSE

0154 04120 016101R JSB SYOUT

0155 04121 126114R JMP ERROR,I

0156 04122 004123R MESSE DEF **1

0157 04123 046040 ASC 2,L

0158* SPACE SINGLE SPACES THE LIST OUTPUT DEVICE

0159* CALLING SEQUENCE:

0160* JSB SPACE

0161*

0162 04125 000000 SPACE NOP

0163 04126 002404 CLA,INA

0164 04127 066132R LDB ANEW

0165 04130 016134R JSB DRKEY

0166 04131 126125R JMP SPACE,I

0167 04132 004133R ANEW DEF **1

0168 04133 020040 ASC 1,

0169* DRKEY PRINTS A MESSAGE ON THE LIST OUTPUT DEVICE

0170* CALLING SEQUENCE:

0171* LDA <# CHARACTERS (POSITIVE)>

0172* LDB <ADDRESS OF MESSAGE>

0173* JSB DRKEY

0174 04134 000000 DRKEY NOP

0175 04135 003004 CMA,INA

0176 04136 073140R STA TEMP

0177 04137 076144R STB **5

0178 04140 016001X JSB EXEC

WRITE ON LIST DEVICE

0179 04141 004146R DEF **5

0180 04142 000055 DEF .2

0181 04143 011157R DEF LISTU

0182 04144 004144R DEF *

0183 04145 011140R DEF TEMP

0184 04146 126134R JMP DRKEY,I

```

0186*      DDOUT WRITES THE CONTENTS OF DBUF ON THE DISC AT THE CURRENT
0187*      DISC SECTOR. FOLLOWING THIS, DBUF IS CLEARED, AND THE CURRENT
0188*      ADDRESS AND COUNT FOR DBUF ARE SET AND THE NEXT DISC ADDRESS IS
0189*      SET INTO DSKAD
0190 04147 000000 DDOUT NOP
0191*
0192 04150 062535R LDA AABUF      IF ABFLG#0, OUTPUT IS FROM
0193 04151 066512R LDB ABFLG      BUFFER A. SET ABFLG # 0 SO
0194 04152 006002  SZB              NEXT OUTPUT WILL BE FROM
0195 04153 062536R LDA ABBUF      BUFFER B. SET CURAD SO
0196 04154 072212R STA EBUF      BUFFER B WILL BE FILLED NEXT
0197 04155 006002  SZB              IF ABFLG#0, OUTUT IS FROM
0198 04156 002400  CLA              BUFFER B. SET ABFLG # 0 SO
0199 04157 072512R STA ABFLG      NEXT OUTPUT WILL BE FROM
0200 04160 062536R LDA ABBUF      BUFFER A. SET CURAD SO
0201 04161 006002  SZB              BUFFER A WILL BE FILLED
0202 04162 062535R LDA AABUF      NEXT.
0203 04163 073033R STA CURAD
0204*
0205 04164 063050R LDA DSKAD      BREAK DOWN DISC ADDR INTO TRACK
0206 04165 017171R JSB GTS          SECTOR FOR EXEC WRITE
0207*
0208 04166 063050R LDA DSKAD      INCREMENT DISC ADDRESS
0209 04167 016221R JSB DISKA      FOR NEXT TIME
0210 04170 073050R STA DSKAD
0211*
0212 04171 063272R LDA N128      INITIALIZE BUFFER COUNT
0213 04172 073044R STA DCNT      TO -128
0214*
0215 04173 062220R LDA FIRST      IS THIS THE 1ST TIME THROUGH?
0216 04174 002002  SZA
0217 04175 026202R JMP NFRST
0218 04176 060055  LDA .2          YES - SET DISC LU = 2 FOR
0219 04177 072507R STA DLU          SYSTEM DISC
0220 04200 072220R STA FIRST      SETFOR NOT 1ST TIME
0221 04201 026204R JMP CL          BYPASS WAIT
0222*
0223 04202 017154R NFRST JSB WAIT      WAIT FOR CURRENT OPERATION
0224 04203 037063R ISZ DSCNT      INCREMENT # SECTORS OUTPUT
0225*
0226 04204 067033R CL LDB CURAD      CLEAR NEXT BUFFER TO BE
0227 04205 016266R JSB BUFCL      FILLED
0228*
0229 04206 016001X JSB EXEC      WRITE RELOCATABLE RECORD
0230 04207 004216R DEF **7
0231 04210 000051  DEF N2          ONTO SYSTEM SCRATCH
0232 04211 004217R DEF WCMND      AREA - WITHOUT WAIT
0233 04212 000000  EBUF NOP          (FILLED FROM ABOVE)
0234 04213 007273R DEF .128
0235 04214 005203R DEF TR
0236 04215 005202R DEF SECT
0237*
0238 04216 126147R JMP DDOUT,I
0239*
0240 04217 020002  WCMND OCT 20002  WRITE WITHOUT AIT COMMAND
0241 04220 000000  FIRST NOP

```

```

0242*      DISKA INCREMENTS THE CURRENT DISC ADDRESS TO
0243*      PROVIDE THE ADDRESS OF THE SUCCEEDING SECTOR.
0244*      CALLING SEQUENCE:
0245*          LDA <CURRENT DISC ADDRESS>
0246*          JSB DISKA
0247*      RETURN:
0248*          A CONTAINS NEXT DISC ADDRESS
0249  04221 000000 DISKA NOP
0250  04222 002004      INA
0251  04223 070001      STA B
0252  04224 010074      AND M377
0253  04225 050116      CPA SECTR      CHECK IF LAST SECTOR ON TRACK
0254  04226 026231R     JMP *+3      INCR TRACK #
0255  04227 060001      LDA B
0256  04230 126221R     JMP DISKA,I   RETURN
0257  04231 060001      LDA B
0258  04232 010075      AND M1774
0259  04233 043267R     ADA M400      INCR TRACK #
0260  04234 126221R     JMP DISKA,I   RETURN
0261*
0262*      DISKI READS A SECTOR OF DATA INTO CORE FROM DISC
0263*      CALLING SEQUENCE:
0264*          LDA <DISC T/S ADDRESS>
0265*          LDB <CORE ADDRESS>
0266*          JSB DISKI
0267*
0268  04235 000000 DISKI NOP
0269  04236 076257R     STB COREA      SAVE CORE BUFF ADDR IN EXEC CALL
0270  04237 017171R     JSB GTS        GET TRACK/SECTOR OF DISC ADDR
0271  04240 003400      CCA            SET FOR READ
0272  04241 016251R     JSB I.O        PERFORM I/O
0273  04242 126235R     JMP DISKI,I    RETURN
0274*
0275*      DISKO WRITES A SECTOR OF DATA ONTO DISC FROM CORE
0276*      CALLING SEQUENCE:
0277*          LDA <DISC ADDRESS>
0278*          LDB <CORE ADDRESS>
0279*          JSB DISKO
0280*
0281  04243 000000 DISKO NOP
0282  04244 076257R     STB COREA      SAVE CORE BUFF ADDR IN EXEC CALL
0283  04245 017171R     JSB GTS        GET TRACK/SECTOR OF DISC ADDR
0284  04246 060051      LDA N2         SET FOR WRITE
0285  04247 016251R     JSB I.O        PERFORM I.O
0286  04250 126243R     JMP DISKO,I    RETURN
0287*
0288  04251 000000 I.O  NOP
0289  04252 072264R     STA RQ          SAVE REQUEST CODE
0290*
0291  04253 016001X     JSB EXEC        EXEC CALL FOR DISC I/O
0292  04254 004263R     DEF *+7
0293  04255 004264R     DEF RQ
0294  04256 011046R     DEF DISCX
0295  04257 000000 COREA NOP
0296  04260 004266R     DEF #WRDS
0297  04261 005203R     DEF TR

```

```

0298 04262 005202R      DEF SECT
0299 04263 126251R      JMP I,0,I
0300*
0301 04264 000000      RQ      NOP      REQUEST CODE*READ*=1; WRITE*=2
0302 04265 000200      #WRDS DEC 128      MAY BE CHANGED BY BLOCK ROUTINE
0303*
0304*      BUFCL CLEARS A 128 WORD BUFFER WITH ZEROS
0305*      CALLING SEQUENCE:
0306*      LDB <BUFFER ADDRESS>
0307*      JSB BUFCL
0308*
0309 04266 000000      BUFCL NOP
0310 04267 063272R      LDA N128
0311 04270 073150R      STA WDCNT      SET CLEAR WORD COUNT
0312 04271 002400      CLA
0313 04272 170001      STA B,I      CLEAR 1 WORD OF BUFFER
0314 04273 006004      INB
0315 04274 037150R      ISZ WDCNT      ALL WORDS CLEARED
0316 04275 026272R      JMP *-3      NO, SO GO CLEAR NEXT WORD
0317 04276 126266R      JMP BUFCL,I      YES, SO RETURN
0318*
0319*      CLID3 CLEARS USAGE FLAGS TO INSURE THAT PROGRAMS WILL BE
0320*      RELOADED AGAIN IF CALLED MORE THAN ONE. THIS IS ESSENTIAL FOR
0321*      ALL LIBRARY AND USER SUBROUTINES. USER MAIN PROGRAMS HOWEVER
0322*      MUST NOT BE CLEARED. BOTH THE ESEAGE FLAG IN THE IDENT ENTRY
0323*      AND THE BP LINK ADDRESS FOR ALL ENTRY POINTS IN THE PROGRAM ARE
0324*      CLEARED.
0325*
0326 04277 000000      CLID3 NOP
0327 04300 060060      LDA .5
0328 04301 067304R      LDB PLAD      GET ADDRESS OF PROGRAM LIST
0329 04302 053121R      CPA PTYPE      PROG = SEGMENT?
0330 04303 067116R      LDB PLMAN      YES-GET MAIN PROG LIST ADDRESS
0331 04304 077036R      STB CURAP      SET CURRENT PROG LIST ADDRESS
0332 04305 007004      CMB,INB
0333 04306 047304R      ADB PLAD      ADD PROGRAM LIST ADDRESS
0334 04307 047273R      ADB .128      ADJUST FOR PROG LIST LENGTH
0335 04310 007000      CMB
0336 04311 077113R      STB PLCNT      SET CURRENT PROG LIST COUNT
0337 04312 163036R      TRID3 LDA CURAP,I      GET CURRENT PROG LIST ENTRY
0338 04313 002003      SZR,RSS      SKIP=NOT ENT OF PROGLIST
0339 04314 126277R      JMP CLID3,I      RETURN=DONE
0340 04315 037036R      ISZ CURAP      INCR CURRENT PROG LIST ADDR
0341 04316 037113R      ISZ PLCNT      SKIP=END OF PROG LIST
0342 04317 002001      RSS
0343 04320 126277R      JMP CLID3,I      RETURN=DONE
0344 04321 073003R      STA TIDNT      SET CURRENT IDENT ADDR
0345 04322 016035R      JSB IDX      SET IDENT ADDRESSES
0346 04323 026637R      JMP RCERR
0347 04324 163013R      LDA ID6,I      GET M/S , TYPE
0348 04325 002020      SSA
0349 04326 026312R      JMP TRID3      IGNORE MAIN
0350 04327 163010R      LDA ID3,I      CLEAR USAGE FLAG
0351 04330 010075      AND M1774
0352 04331 173010R      STA ID3,I
0353 04332 016000R      JSB INLST      INITIALIZE LSTX

```

0354	04333	016004R	CLSUT	JSB LSTX	SET CURRENT LST ADDRESSES
0355	04334	026312R		JMP TRID3	TRY NEXT IDENT
0356	04335	163000R		LDA LST4,I	SET IDENT ADDRESS
0357	04336	053006R		CPA ID1	ENT/EXT BELONGS TO CURRENT PROG
0358	04337	006401		CLB,RSS	YES
0359	04340	026333R		JMP CLSUT	TRY NEXT LST ENTRY
0360	04341	177001R		STB LST5,I	CLEAR BP LINK ADDRESS
0361	04342	026333R		JMP CLSUT	TRY NEXT LST ENTRY
0362*					
0363*	BPSCN SCANS LOADER BP AREA FOR A VALUE EQUAL TO THE CURRENT				
0364*	OPERAND. IF A VALUE IS FOUND, THE ADDRESS OF THE OPERAND IS				
0365*	RETURNED IN THE A REGISTER. OTHERWISE A NEW BP LINK WORD IS				
0366*	RESERVED AND THE ADDRESS OF THIS WORD IS RETURNED IN A.				
0367*					
0368	04343	000000	BPSCN	NOP	
0369	04344	063152R		LDA UBFWA	LOW BASE PAGE ADDRESS
0370	04345	073101R		STA LOWER	SAVE LOWER BOUND
0371	04346	067135R		LDB TBREL	HIGH BASE PAGE ADDRESS
0372	04347	007004		CMB,INB	
0373	04350	077145R		STB UPPER	SAVE NEG UPPER BOUND
0374	04351	063101R	BPSRC	LDA LOWER	GET CURRENT LOW BP BOUND
0375	04352	043145R		ADA UPPER	TEST FOR END OF AREA
0376	04353	002021		SSA,RSS	NOT END OF AREA
0377	04354	026367R		JMP BPALC	GO TO ALLOCATE LOCATION
0378	04355	067105R		LDB OPRND	GET OPERAND
0379	04356	063101R		LDA LOWER	
0380	04357	043026R		ADA BPOIT	
0381	04360	043205R		ADA BPBUF	ADJUST FOR START OF BP BUFFER
0382	04361	037101R		ISZ LOWER	INCR LOWER BOUND
0383	04362	164000		CPB A,I	CHECK IF OPERANDS SAME
0384	04363	002001		RSS	
0385	04364	026351R		JMP BPSRC	NO-TRY NEXT VALUE
0386	04365	043103R		ADA NBPBF	READJUST FOR BP BUFFER ADDR
0387	04366	126343R		JMP BPSCN,I	RETURN WITH ADDR IN A
0388	04367	017011R	BPALC	JSB DLINK	GO CHECK FOR CURRENT PAGE LINK
0389	04370	016444R		JSB ALLOC	ALLOCATE NEW BP LINK ADDRESS
0390	04371	126343R		JMP BPSCN,I	RETURN WITH ADDR IN A
0391*					
0392*	LSTEX LISTS THE UNDEFINED EXTERNAL REFERENCES				
0393*	CALLING SEQUENCE:				
0394*	JSB LSTEX				
0395*					
0396	04372	000000	LSTEX	NOP	
0397	04373	002400		CLA	
0398	04374	073065R		STA EXCNT	CLEAR UNDEFINED EXT COUNT
0399	04375	016000R		JSB INLST	
0400	04376	016004R	LST10	JSB LSTX	
0401	04377	026416R		JMP LST20	DONE LISTING UNDEFINED EXT
0402	04400	163000R		LDA LST4,I	
0403	04401	002021		SSA,RSS	
0404	04402	026376R		JMP LST10	NOT UNDEFINED, SO GET NEXT
0405	04403	063065R		LDA EXCNT	CHECK IF UNDEFINED EXT HEADING
0406	04404	002002		SZA	PUT OUT
0407	04405	026411R		JMP ++4	YES
0408	04406	063172R		LDA .14	NO
0409	04407	066423R		LDB MESUX	

```

0410 04410 016101R JSB SYOUT OUTPUT HEADING
0411 04411 060060 LDA ,5
0412 04412 066775R LDB LST1
0413 04413 016101R JSB SYOUT
0414 04414 037065R ISZ EXCNT INCR UNDEFINED EXT COUNT
0415 04415 026376R JMP LST10 GET NEXT LST
0416 04416 063065R LST20 LDA EXCNT CHECK IF ANY UNDEFINED EXTERNALS
0417 04417 002002 SZA SKIP=NO
0418 04420 126372R JMP LSTEX,I RETURN TO P+1 -UNDEFINED EXTERNAL
0419 04421 036372R ISZ LSTEX INCR RETURN ADDRESS
0420 04422 126372R JMP LSTEX,I RETURN TO P+2
0421 04423 004424R MESUX DEF **1
0422 04424 052516 ASC 7,UNDEFINED EXT8
0423*
0424* SETOP SETS CURRENT OPERAND IN TH BP LOCATION SPECIFIED
0425* BY TH ADDRESS IN A
0426*
0427 04433 000000 SETOP NOP
0428 04434 072443R STA BPADD
0429 04435 043026R ADA BPOIT
0430 04436 043205R ADA BPBUF ADJUST BP LINK ADDRESS
0431 04437 067105R LDB OPRND GET AND STORE OPERAND
0432 04440 174000 STB A,I INTO BP LOCATION
0433 04441 062443R LDA BPADD
0434 04442 126433R JMP SETOP,I RETURN
0435 04443 000000 BPADD NOP
0436*
0437* ALLOC ESTABLISHES ALL THE BASE PAGE LINKAGE ADDRESSES.
0438* DIAGNOSTIC IS OUTPUT IF OUT OF SPACE.
0439*
0440 04444 000000 ALLOC NOP
0441 04445 063135R LDA TBREL GET NEXT BP ADDRESS
0442 04446 043026R ADA BPOIT
0443 04447 043205R ADA BPBUF ADJUST BP ADDRESS
0444 04450 006400 CLB
0445 04451 174000 STB A,I CLEAR ALLOCATED BP LOCATION
0446 04452 063135R LDA TBREL
0447 04453 037135R ISZ TBREL INCR NEXT BP ADDRESS
0448 04454 064000 LDB A CHECK IF BASE PAGE OVERFLOW
0449 04455 007000 CMB
0450 04456 047153R ADB UBLWA
0451 04457 006021 SSB,RSS
0452 04460 126444R JMP ALLOC,I
0453 04461 063166R LDA ERR04 BP LINK OVERFLOW ERROR
0454 04462 026370R JMP ABRTX ABORT LOADR
0455*
0456* DBSET GIVES THE ADDRESS OF THE NEXT WORD OF THE RELOCATABLE
0457* RECORD IN LBUF. IF AT END OF BUFFER, IT READS IN ANOTHER PACKED
0458* RELOCATABLE RECORD
0459*
0460 04463 000000 DBSET NOP
0461 04464 037035R ISZ CURAL INCR CURRENT LBUF ADDRESS
0462 04465 037077R ISZ LCNT SKIP=END OF LBUF
0463 04466 126463R JMP DBSET,I RETURN
0464 04467 017137R JSB INREL READ NEXT RELOCAABLE RECORD
0465 04470 126463R JMP DBSET,I RETURN

```

```

0467*
0468* INPUT NAM RECORD FROM DISC
0469*
0470 04471 000000 INNAM NOP
0471 04472 073055R STA DSKRD SAVE ORIGINAL DISC ADDRESS
0472*
0473 04473 063046R LDA DISCX
0474 04474 072507R STA DLU DEPENDING ON DISCX
0475*
0476 04475 032510R IOR WWC SET UP READ COMMAND
0477 04476 072511R STA CMND WITHOUT WAIT
0478*
0479 04477 002400 CLA CLEAR # SECTORS
0480 04500 073063R STA DSCNT READ
0481 04501 072512R STA ABFLG SET FOR BUFFER A
0482*
0483 04502 063055R LDA DSKRD BREAK DOWN DISC ADDRESS FOR
0484 04503 017171R JSB GTS READ
0485*
0486 04504 016513R JSB DRWOW ISSUE READ WITHOUT WAIT FOR BU A
0487*
0488 04505 017137R JSB INREL SET FOR B, WAIT FOR A, REQUEST B
0489*
0490 04506 126471R JMP INNAM,I
0491*
0492 04507 000000 DLU NOP DISC LOGICAL UNIT #
0493 04510 020000 WWC OCT 20000 WITHOUT WAIT
0494 04511 000000 CMND NOP
0495 04512 000000 ABFLG NOP A/B BUFFER FLAG

```

```

0497*
0498* DISK READ WITHOUT WAIT
0499*
0500 04513 000000 DRWOW NOP
0501*
0502 04514 062535R LDA AABUF IF ABFLG#0, SET FOR INPUT
0503 04515 066512R LDB ABFLG INTO AABUF THEN SET
0504 04516 006002 SZB ABFLG#0.
0505 04517 062536R LDA ABBUF IF ABFLG#0, SET FOR INPUT
0506 04520 072530R STA CBUF INTO BBUF, THEN SET
0507 04521 006002 SZB ABFLG#0.
0508 04522 002400 CLA
0509 04523 072512R STA ABFLG
0510*
0511 04524 016001X JSB EXEC
0512 04525 004534R DEF ++7
0513 04526 000052 DEF N1
0514 04527 004511R DEF CMND
0515 04530 000000 CBUF NOP SET FROM ABFLG
0516 04531 007273R DEF .128
0517 04532 005203R DEF TR
0518 04533 005202R DEF SECT
0519*
0520 04534 126513R JMP DRWOW,I

```

0521*

```

0522 04535 004537R AABUF DEF ABUF
0523 04536 004737R ABBUF DEF BBUF
0524 04537 000000 ABUF BSS 128
0525 04737 000000 BBUF BSS 128

```

0527*

0528* INPUT RELOCATABLE RECORD

0529*

0530 05137 000000 INREL NOP

0531*

```

0532 05140 063055R LDA DSKRD INCREMENT DISC
0533 05141 017204R JSB NDA COMPUTE NEXT DISC ADDRESS
0534 05142 073055R STA DSKRD BREAK INTO TRACK/SECTOR
0535 05143 017171R JSB GTS

```

0536*

0537 05144 017154R JSB WAIT

0538*

```

0539 05145 062530R LDA CBUF INITIALIZE RELOCATABLE INPUT
0540 05146 073035R STA CURAL BUFFER POINTERS

```

0541 05147 063272R LDA N128

0542 05150 073077R STA LCNT

0543*

0544 05151 037063R ISZ DSCNT INCREMENT # SECTOR READ

0545*

0546 05152 016513R JSB DRWOW INITIATE NEXT READ

0547*

0548 05153 127137R JMP INREL,I

0550*

0551* WAIT FOR CURRENT DISC TRANSFER TO COMPLTE

0552*

0553 05154 000000 WAIT NOP

0554*

0555 05155 016001X JSB EXEC

0556 05156 005163R DEF **5

0557 05157 005166R DEF N13

0558 05160 004507R DEF DLU

0559 05161 005167R DEF STAT

0560 05162 005170R DEF TLOG

0561*

0562 05163 002020 SSA

0563 05164 027155R JMP WAIT+1

0564*

0565 05165 127154R JMP WAIT,I

0566*

0567 05166 177763 N13 DEC -13

0568 05167 000000 STAT NOP

0569 05170 000000 TLOG NOP

```

STATUS REQUEST
LOGICAL UNIT SPECIFIED AT NAM

```

```

IF BUSY,
LOOP

```

```

0571*
0572* GET TRACK/SECTOR FROM CURRENT DISC ADDRESS
0573*
0574 05171 000000 GTS NOP
0575*
0576 05172 070001 STA B SAVE CURRENT DISC ADDRESS
0577 05173 010073 AND M177
0578 05174 073202R STA SECT SECTOR # IN LOW 8 BITS
0579 05175 060001 LDA B
0580 05176 001727 ALF,ALF
0581 05177 010074 AND M377
0582 05200 073203R STA TR TRACK # IN UPPER 8 BITS
0583*
0584 05201 127171R JMP GTS,I
0585*
0586 05202 000000 SECT NOP CURRENT DISC SECTOR #
0587 05203 000000 TR NOP CURRENT DISC TRACK #

0589*
0590* NEXT DISC ADDRESS
0591*
0592 05204 000000 NDA NOP
0593*
0594 05205 016221R JSB DISKA ADDRESS NORMALLY
0595*
0596 05206 067137R LDB TDIR IF FORWARD INCREMNT,
0597 05207 006003 SZB,RSS NEW DISC ADDRESS IS
0598 05210 127204R JMP NDA,I IN "A"
0599*
0600 05211 070001 STA B IF BACKWARD INCREMENT,
0601 05212 010073 AND M177 FIRST CHECK TO SEE
0602 05213 002002 SZA IF NEW TRACK
0603 05214 027220R JMP NDA2 NO- VALUE IN B IS OK
0604*
0605 05215 005727 BLF,BLF YES- DECREMENT BACK 2
0606 05216 044051 ADB N2 TRACKS
0607 05217 005727 BLF,BLF ROTATE TO HIGH 8 BITS
0608*
0609 05220 060001 NDA2 LDA B UPDATE NW
0610*
0611 05221 127204R JMP NDA,I

```

```

0613*
0614*      GENID GENERATEES THE DIRECTORY ENTRY FOR THE PROGRAM LOADED.
0615*
0616 05222 000000 GENID NOP
0617 05223 003400 CCA
0618 05224 073114R STA PLFLG
0619 05225 063074R LDA IMAIN      SET ADDRESS FOR IDX
0620 05226 073003R STA TIDNT
0621 05227 016035R JSB IDX
0622 05230 026637R JMP RCERR
0623 05231 163006R LDA ID1,I
0624 05232 017421R JSB OUTID      NAME 1,2 TO DIRECTORY
0625 05233 163007R LDA ID2,I
0626 05234 017421R JSB OUTID      NAME 3,4 TO DIRECTORY
0627 05235 163013R LDA ID6,I      GET TYPE
0628 05236 010073 AND M177
0629 05237 067156R LDB DVR        LOADING DRIVERS?
0630 05240 054057 CPB .4
0631 05241 060057 LDA .4          YES - RESET TO TYPE 4
0632 05242 033265R IOR M200      SET -P- BIT
0633 05243 070001 STA B
0634 05244 163010R LDA ID3,I      NAME 5
0635 05245 010075 AND M1774
0636 05246 030001 IOR B
0637 05247 017421R JSB OUTID      NAME 5,P,TYPE TO DIRECTORY
0638 05250 063054R LDA DSKMN
0639 05251 017421R JSB OUTID      MAIN DISC ADDR TO DIRECTORY
0640 05252 063117R LDA PPREL
0641 05253 067121R LDB PTYPE
0642 05254 054060 CPB .5
0643 05255 063022R LDA B$PAD
0644 05256 043023R ADA B$SDP
0645 05257 072210R STA TBUF
0646 05260 003004 CMA,INA
0647 05261 043323R ADA TPREL
0648 05262 043175R ADA .127
0649 05263 013270R AND M7600
0650 05264 001727 ALF,ALF
0651 05265 001200 RAL
0652 05266 072211R STA TBUF+1     MAIN # OF SECTORS
0653 05267 063107R LDA PBREL
0654 05270 054060 CPB .5
0655 05271 063021R LDA B$BAD
0656 05272 072212R STA TBUF+2     LOW BASE
0657 05273 003004 CMA,INA
0658 05274 043135R ADA TBREL
0659 05275 043175R ADA .127
0660 05276 013270R AND M7600
0661 05277 001727 ALF,ALF
0662 05300 001200 RAL
0663 05301 042211R ADA TBUF+1     TOTAL # OF SECTORS
0664 05302 017421R JSB OUTID      NUMBER OF SECTORS TO DIRECTORY
0665 05303 062210R LDA TBUF
0666 05304 017421R JSB OUTID      LOW MAIN ADDR TO DIRECTORY
0667 05305 063323R LDA TPREL
0668 05306 017421R JSB OUTID      HIGH MAIN ADDR TO DIRECTORY

```

0669	05307	062212R	LDA TBUF+2	
0670	05310	017421R	JSB OUTID	LOW BASE ADDRESS TO DIRECTORY
0671	05311	063135R	LDA TBREL	
0672	05312	017421R	JSB OUTID	HIGH BASE ADDR TO DIRECTORY
0673*				
0674	05313	067156R	LDB DVR	IF LOADING I/O DRIVERS, FIND
0675	05314	054057	CPB .4	I.XX AND C.XX ENTRY POINTS
0676	05315	017347R	JSB DVRAD	FOR DIRECTORY ENTRY
0677*				
0678	05316	063120R	LDA PRENT	
0679	05317	017421R	JSB OUTID	PRIMARY ENTRY ADDRESS TO DIRECT
0680	05320	063346R	LDA LIBRY	
0681	05321	002003	SZA,RSS	
0682	05322	063323R	LDA TPREL	
0683	05323	017421R	JSB OUTID	LAST WORD ADDRESS TO DIRECTORY
0684	05324	067071R	LDB ICNT	CHECK IF BEGINNING OF NEW SECTRR
0685	05325	057272R	CPB N128	
0686	05326	127222R	JMP GENID,I	NEW SECTOR BEGUN RETURN
0687	05327	063071R	LDA ICNT	CHECK IF ROOM FOR ANOTHER
0688	05330	040065	ADA .10	DIRECTORY ENTRY
0689	05331	002020	SSA	
0690	05332	127222R	JMP GENID,I	ROOM FOR ANOTHER ENTRY RETURN
0691	05333	003400	CCA	
0692	05334	017421R	JSB OUTID	SET ANOTHER DIRECTORY ENTRY FOLLO
0693	05335	067071R	LDB ICNT	
0694	05336	077140R	STB TEMP	SET ZERO OUTPUT COUNT
0695	05337	057272R	CPB N128	
0696	05340	127222R	JMP GENID,I	
0697	05341	002400	CLA	
0698	05342	017421R	JSB OUTID	OUTPUT ZERO TO DIR BUFFER
0699	05343	037140R	ISZ TEMP	
0700	05344	027341R	JMP *-3	OUTPUT ANOTHER ZERO TO BUFFER
0701	05345	127222R	JMP GENID,I	RETURN
0702	05346	000000	LIBRY NOP	FIRST LIBR PROG LOAD ADDRESS
0703*				
0704*	DVRAD			SEARCHES THE LST FOR THE I.XX & C.XX
0705*				ENTRY POINTS FOR THE I/O DRIVER BEING
0706*				LOADED.
0707*				'PRENT' IS SET TO ADDRESS OF I.XX
0708*				'LIBRY' IS SET TO ADDRESS OF C.XX
0709*				THESE VALUES ARE THEN STORED IN
0710*				WORDS 10 & 11 OF THE DIRECTORY ENTRY
0711*				
0712	05347	000000	DVRAD NOP	
0713	05350	163007R	LDA ID2,I	FIND THE XX NUMBER FROM
0714	05351	010073	AND M177	THE DVRXX NAME. STORE
0715	05352	001727	ALF,ALF	THE ASCII XX IN TEMP.
0716	05353	073140R	STA TEMP	
0717	05354	163010R	LDA ID3,I	
0718	05355	001727	ALF,ALF	
0719	05356	010073	AND M177	
0720	05357	033140R	IOR TEMP	
0721	05360	073140R	STA TEMP	
0722*				
0723	05361	016000R	JSB INLST	SEARCH LST
0724	05362	016004R	DVR1 JSB LSTX	

0725	05363	127347R	JMP	DVRAD,I	
0726	05364	162775R	LDA	LST1,I	
0727	05365	053417R	CPA	IDEC	I. ?
0728	05366	027375R	JMP	IP	
0729	05367	053420R	CPA	CDEC	C. ?
0730	05370	002001	RSS		
0731	05371	027362R	JMP	DVR1	
0732	05372	017400R	JSB	DVRS	CHECK FOR CORRECT XX
0733	05373	073346R	STA	LIBRY	SET C,XX ADDRESS
0734	05374	027362R	JMP	DVR1	
0735*					
0736	05375	017400R	IP	JSB	DVRS
0737	05376	073120R	STA	PRENT	CHECK FOR CORRECT XX
0738	05377	027362R	JMP	DVR1	SET I,XX ADDRESS
0739*					
0740*					
0741	05400	000000	DVRS	NOP	
0742	05401	162776R	LDA	LST2,I	XX MATCH?
0743	05402	053140R	CPA	TEMP	
0744	05403	002001	RSS		
0745	05404	027362R	JMP	DVR1	NO
0746	05405	162777R	LDA	LST3,I	
0747	05406	010075	AND	M1774	
0748	05407	053211R	CPA	UBLNK	5TH CAHR = BLANK?
0749	05410	002001	RSS		
0750	05411	027362R	JMP	DVR1	NO
0751	05412	163001R	LDA	LST5,I	GET BP LINK
0752	05413	043026R	ADA	BPOIT	COMPUTE ABSOLUTE ADDRESS
0753	05414	043205R	ADA	BPBUF	
0754	05415	160000	LDA	A,I	
0755	05416	127400R	JMP	DVRS,I	RETURN WITH ADDRESS IN "A"
0756*					
0757	05417	044456	IDEC	ASC	1,I.
0758	05420	041456	CDEC	ASC	1,C.
0759*					
0760*					
0761*	OUTID STORES DIRECTORY ENTRIES IN ID SEGMENT BUFFER				
0762*	AND WRITES IT ON DISC WHEN IT BECOMES FULL.				
0763*					
0764	05421	000000	OUTID	NOP	
0765	05422	173034R	STA	CURAI,I	STORE WORD INTO ID BUFFER
0766	05423	037034R	ISZ	CURAI	INCR BUFFER ADDR
0767	05424	037071R	ISZ	ICNT	INCR COUNT
0768	05425	127421R	JMP	OUTID,I	RETURN=NOT FULL
0769	05426	017537R	JSB	SETCR	SET USER DISC FLAG IN BASE PAGE
0770	05427	063053R	LDA	DSKID	
0771	05430	067275R	LDB	AIBUF	
0772	05431	016243R	JSB	DISKO	WRITE 1 SECTOR OF DIRECTORY
0773	05432	067275R	LDB	AIBUF	
0774	05433	077034R	STB	CURAI	SET NEW CURRENT IBUF ADDR
0775	05434	016266R	JSB	BUFCL	CLEAR BUFFER
0776	05435	060174	LDA	TSONE	CURRENT DISC ADDRESS
0777	05436	073053R	STA	DSKID	
0778	05437	010073	AND	M177	
0779	05440	002003	SZA	RSS	NO
0780	05441	026320R	JMP	DRCHA	OUT OF DIRECTORY SPACE

0781	05442	063272R	LDA N128	
0782	05443	073071R	STA ICNT	SET CURRENT IBUF COUNT
0783	05444	127421R	JMP OUTID,I	RETURN

```

0785*      LABDO PUTS OUT CURRENT ABSOLUTE CODE WORD FOR THE PROGRAM
0001*      BEING LOADED.
0002*      IF WORD IS IN CURRENT SECTOR, IT IS INSERTED.
0003*      IF WORD DOES NOT FIT IN CURRENT SECTOR, THE CURRENT SECTOR
0004*      IS OUTPUT AND THE NEW SECTOR READ IN.
0005*      ADDITIONALLY, THERE ARE CHECKS FOR MEMORY OVERFLOW AND MEMORY
0006*      OVERLAP
0007*      CALLING SEQUENCE:
0008*      LDA <CURRENT ABSOLUTE CODE WORD>
0009*      LDB <CURRENT ABSOLUTE ADDRESS>
0010*      JSB LABDO
0011 05445 000000 LABDO NOP
0012 05446 073133R STA TABS      ABSOLUTE CODE WORD
0013 05447 077134R STB TADR      ABSOLUTE CORE ADDRESS
0014 05450 077123R STB RELAD     UPDATE RELOCATION ADDRESS
0015 05451 007004  CMB,INB
0016 05452 047155R ADB UMLWA     CHECK IF MEMORY OVERFLOW
0017 05453 006021  SSB,RSS
0018 05454 027457R JMP LAB10     NO
0019 05455 063165R LDA ERR03     YES-80 OUTPUT MESSAGE
0020 05456 026370R JMP ABRTX      AND ABORT
0021 05457 063027R LAB10 LDA CBASE  COMPUTE # OF WORDS
0022 05460 003004  CMA,INA
0023 05461 043134R ADA TADR
0024 05462 016024R JSB RRL9
0025 05463 073144R STA TWORD     WORD #
0026 05464 007004  CMB,INB
0027 05465 077136R STB TCNT
0028 05466 063040R LDA DBASE
0029 05467 006003  SZB,RSS
0030 05470 027474R JMP ++4
0031 05471 016221R JSB DISKA
0032 05472 037136R ISZ TCNT
0033 05473 027471R JMP *-2
0034 05474 073142R STA TNEXT     ABSLLUTE TRACK/SECTOR #
0035 05475 063045R CPA DCRNT     CHECK IF CURRENT SECTOR
0036 05476 027503R JMP LAB20     YES
0037 05477 017654R JSB REMDO     NO- WRITE OUT CURRENT SECTOR
0038 05500 063142R LDA TNEXT
0039 05501 073045R STA DCRNT
0040 05502 017550R JSB CLRTK     SNOOWFLOW TRACKS
0041 05503 067274R LAB20 LDB ADBUF  COMPUTE ADDRESS TO STORE WORD
0042 05504 047144R ADB TWORD
0043 05505 063133R LDA TABS      CHECK IF ABS WORD IS HALT 0
0044 05506 002003  SZA,RSS
0045 05507 027527R JMP LAB40     YES-80 DON'T STORE
0046 05510 160001  LDA B,I       SO-80 CHECK IF WORD IN BUFFER
0047 05511 002003  SZA,RSS      IS 0
0048 05512 027525R JMP LAB30     YES-80 GO STORE
0049 05513 063043R LDA DBUGF     CHECK IF DEBUG FLAG SET
0050 05514 002002  SZA          NO-80 MEMORY OVERLAY
0051 05515 027523R JMP LAB25     YES
0052 05516 063115R LDA PLGTH     CHECK IF COMPILER PRODUCED PROG
0053 05517 002020  SSA          SKIP-ASSEMBLER PRODUCED
0054 05520 027525R JMP LAB30
0055 05521 063202R LDA ERR16     NO-80 PROGRAM OVERLAY

```

0056	05522	026370R	JMP ABRTX	
0057	05523	002400	LAB25 CLA	
0058	05524	073043R	STA DBUGF	CLEAR DEBUG FLAG
0059	05525	063133R	LAB30 LDA TABS	
0060	05526	170001	STA B,I	STORE WORD IN BURFER
0061	05527	063123R	LAB40 LDA RELAD	CHECK IF NEW MAXIMUM ADDRESS
0062	05530	070001	STA B	
0063	05531	003004	CMA,INA	
0064	05532	043102R	ADA MAXAD	
0065	05533	002021	SSA,RSS	
0066	05534	127445R	JMP LABD0,I	RETURN
0067	05535	077102R	STB MAXAD	SET NEW MAX ADDRESS
0068	05536	127445R	JMP LABD0,I	RETURN

0070	05537	000000	SETCR	NOP	SET USER DISC FLAG IN BASE PAGE
0071	05540	060056		LDA ,3	
0072	05541	073046R		STA DISCX	
0073	05542	127537R		JMP SETCR,I	
0074*					
0075*	BASE	PAGE	STORE		
0076*					
0077	05543	000000	BPSTR	NOP	STORE A IN B
0078	05544	016001X		JSB EXEC	
0079	05545	005547R		DEF **2	
0080	05546	007167R		DEF N19	
0081	05547	127543R		JMP BPSTR,I	
0082*					
0083*	CLEAR	TRACKS			
0084*					
0085	05550	000000	CLRTK	NOP	
0086	05551	001727		ALF,ALF	RIGHT JUSTIFY
0087	05552	010074		AND M377	TO LOW 8 BITS
0088	05553	043057R		ADA DSKTM	COMPARE TO UPPR TRACK LIMIT
0089	05554	002021		SSA,RSS	NOT TO BIG IF NEG
0090	05555	026320R		JMP DRCHA	ERROR L10
0091	05556	063045R		LDA DCRNT	CURRENT REQUEST T/S
0092	05557	010075		AND M1774	
0093	05560	001727		ALF,ALF	RIGHT JUSTIFY
0094	05561	070001		STA B	B = TRACK
0095	05562	063045R		LDA DCRNT	CURRENT REQUEST T/S
0096	05563	010074		AND M377	A = SECTOR
0097	05564	047060R		ADB DSK01	*,S COMP OF LAST TRACK CLEARED
0098	05565	006020		SSB	ALREADY ZEROED IF NEG
0099	05566	027624R		JMP LAB19	
0100	05567	006002		SZB	
0101	05570	027576R		JMP UPDAT	WRITE ZEROS ON DISC UP TO TNEXT
0102	05571	043061R		ADA DSK02	CPMPARE SECTOR COUNTS
0103	05572	002020		SSA	
0104	05573	027624R		JMP LAB19	ALREADY ZEROED IF NEG
0105	05574	002003		SZA,RSS	0 IS OK; > 0 NEEDS UPDATE
0106	05575	027624R		JMP LAB19	
0107	05576	067274R	UPDAT	LDB ADBUF	CLEAR 128 WD BUFFER (DBUF)
0108	05577	016266R		JSB BUFCL	
0109	05600	017537R		JSB SETCR	
0110	05601	063062R	UPDA1	LDA DSKUD	
0111	05602	067274R		LDB ADBUF	
0112	05603	016243R		JSB DISKO	WRITE SECTOR OF 0,S ON DISC
0113	05604	063062R		LDA DSKUD	
0114	05605	064174		LDB TSONE	UPDATE NEXT T/S NOT CLEARED
0115	05606	077062R		STB DSKUD	
0116	05607	070001		STA B	
0117	05610	053045R		CPA DCRNT	DONE IF COMPARE
0118	05611	027614R		JMP **3	
0119	05612	017537R		JSB SETCR	
0120	05613	027601R		JMP UPDA1	
0121	05614	010075		AND M1774	
0122	05615	001727		ALF,ALF	
0123	05616	003004		CMA,INA	
0124	05617	073060R		STA DSK01	COMP OF LAST TRACK CLEARED
0125	05620	060001		LDA B	

```

0126 05621 010074      AND M377
0127 05622 003004      CMA,INA
0128 05623 073061R     STA DSK02      COMP OF LAST SECTOR CLEARED
0129 05624 017537R LAB19 JSB SETCR
0130 05625 063045R     LDA DCRNT
0131 05626 067274R     LDB ADBUF
0132 05627 016235R     JSB DISKI      READ IN DESIRED SECTOR
0133 05630 127550R     JMP CLRTK,I
0134*
0135* INITIALIZE 1ST DISC TRACK
0136*
0137 05631 000000      INDSK NOP
0138 05632 073654R     STA REMDO      SAVE TEMP
0139 05633 010075      AND M1774     GET START TRAK NUM
0140 05634 001727      ALF,ALF       RIGHT JUSTIFY
0141 05635 003004      CMA,INA       MAKE NEG
0142 05636 073060R     STA DSK01
0143 05637 063654R     LDA REMDO
0144 05640 010074      AND M377      GET START SECTOR NUM
0145 05641 003004      CMA,INA
0146 05642 073061R     STA DSK02
0147 05643 067274R     LDB ADBUF
0148 05644 016266R     JSB BUFCL
0149 05645 017537R     JSB SETCR
0150 05646 063654R     LDA REMDO
0151 05647 067274R     LDB ADBUF      CORA ADDRESS
0152 05650 016243R     JSB DISKO     WRITE DISC
0153 05651 060174      LDA TSONE     NEXT AVAILABLE T/S
0154 05652 073062R     STA DSKUD
0155 05653 127631R     JMP INDSK,I
0156*
0157* REMDO OUTPUTS CURRENT BUFFER.
0158* NORMALLY USED WHEN CODE FOR MAIN PROGRAM OR SEGMENT HAS BEEN
0159* GENERATED.
0160*
0161 05654 000000      REMDO NOP
0162 05655 017537R     JSB SETCR      SET USER DISC FLAG IN BASE PAGE
0163 05656 063045R     LDA DCRNT
0164 05657 067274R     LDB ADBUF
0165 05660 016243R     JSB DISKO
0166 05661 127654R     JMP REMDO,I
0167* READ IN INITIAL DISC SECTOR
0168 05662 000000      DINIT NOP
0169 05663 063050R     LDA DSKAD      SACE INITIAL DISC ADDRESS
0170 05664 073040R     STA DBASE
0171 05665 073045R     STA DCRNT
0172 05666 017631R     JSB INDSK     INITILIZE START DISC TRAK ADD
0173 05667 002400      CLA
0174 05670 073102R     STA MAXAD     CLEAR MAXIMUM ADDRESS FLAG
0175 05671 127662R     JMP DINIT,I
0176* COMPUTE AND UPDATE DSKAD ADDRESS
0177 05672 000000      RESET NOP
0178 05673 062210R     LDA TBUF
0179 05674 072023R     STA TTBUF
0180 05675 006400      CLB
0181 05676 063030R     LDA CLXXX

```

0182	05677	077030R	STB CLXXX	
0183	05700	002003	SZA,RSS	
0184	05701	026005R	JMP RES05	
0185	05702	016031R	JSB INIDX	
0186	05703	016035R	RES15 JSB IDX	
0187	05704	027751R	JMP RES50	
0188	05705	163006R	LDA ID1,I	CHECK IF IDENT IS -DEBUG-
0189	05706	053217R	CPA CHRDE	
0190	05707	002001	RSS	
0191	05710	027721R	JMP RES20	NO
0192	05711	163007R	LDA ID2,I	
0193	05712	053220R	CPA CHRDE+1	
0194	05713	002001	RSS	
0195	05714	027721R	JMP RES20	NO
0196	05715	163010R	LDA ID3,I	
0197	05716	010075	AND M1774	
0198	05717	053221R	CPA CHRDE+2	
0199	05720	027736R	JMP RES30	YES
0200	05721	163006R	RES20 LDA ID1,I	CHECK IF -DBKPT-
0201	05722	053230R	CPA CHRDB	
0202	05723	002001	RSS	
0203	05724	027703R	JMP RES15	NO
0204	05725	163007R	LDA ID2,I	
0205	05726	053231R	CPA CHRDB+1	
0206	05727	002001	RSS	
0207	05730	027703R	JMP RES15	
0208	05731	163010R	LDA ID3,I	
0209	05732	010075	AND M1774	
0210	05733	053232R	CPA CHRDB+2	
0211	05734	002001	RSS	YES
0212	05735	027703R	JMP RES15	NO
0213	05736	173010R	RES30 STA ID3,I	CLEAR USAGE
0214	05737	067304R	LDB PLAD	CLEAR PLIST LOADED FLAG
0215	05740	160001	LDA B,I	
0216	05741	001665	ELA,CLE,ERA	
0217	05742	053006R	CPA ID1	
0218	05743	027746R	JMP RES40	
0219	05744	006004	INB	
0220	05745	027740R	JMP RES30+2	
0221	05746	002400	RES40 CLA	
0222	05747	170001	STA B,I	
0223	05750	027703R	JMP RES15	
0224	05751	063036R	RES50 LDA CURAP	MOVE PLIST POINTER
0225	05752	040051	ADA N2	BACK 2 ENTRIES
0226	05753	073036R	STA CURAP	
0227	05754	060047	LDA N4	
0228	05755	073136R	STA TCNT	
0229	05756	067213R	LDB DADDR	
0230	05757	077140R	STB TEMP	
0231	05760	163140R	RES60 LDA TEMP,I	
0232	05761	072210R	STA TBUF	SET NAME FOR LSCAN
0233	05762	037140R	ISZ TEMP	
0234	05763	163140R	LDA TEMP,I	
0235	05764	072211R	STA TBUF+1	
0236	05765	037140R	ISZ TEMP	
0237	05766	163140R	LDA TEMP,I	

```

0238 05767 072212R STA TBUF+2
0239 05770 037140R ISZ TEMP
0240 05771 017334R JSB LSCAN SEARCH LST FOR SYMBOL
0241 05772 026003R JMP RES70 NOT FOUND
0242 05773 162777R LDA LST3,I FOUND
0243 05774 010075 AND M1774
0244 05775 172777R STA LST3,I CLEAR ORDINAL
0245 05776 163000R LDA LST4,I
0246 05777 001665 ELA,CLE,ERA
0247 06000 173000R STA LST4,I SET IDENT POSITIVE
0248 06001 002400 CLA
0249 06002 173001R STA LST5,I CLEAR BP LINK ADDRESS
0250 06003 037136R RES70 ISZ TCNT
0251 06004 027760R JMP RES60
0252 06005 062023R RES05 LDA TBUF BASE PROGRAM ADDRESS
0253 06006 003004 CMA,INA
0254 06007 043323R ADA TPREL ADD MAXIMUM ADDRESS FLAG
0255 06010 016024R JSB RRL9
0256 06011 002002 SZA
0257 06012 006004 INB
0258 06013 007004 CMB,INB
0259 06014 077136R STB TCNT NEG # SECTORS USED
0260 06015 063040R LDA DBASE GET STARTING DISC ADDRESS
0261 06016 016221R JSB DISKA INCR # OF SECTORS
0262 06017 037136R ISZ TCNT USED TIME
0263 06020 026016R JMP *-2
0264 06021 073050R STA DSKAD STORE NEXT DISC T/S ADDRESS
0265 06022 127672R JMP RESET,I
0266 06023 000000 TBUF NOP
0267 06024 000000 RRL9 NOP DO RRL 9 TO B+A REG
0268 06025 070001 STA B
0269 06026 013270R AND M7600 MASK OF UPPER 9 BITS
0270 06027 001727 ALF,ALF TAKE UPPER 9 BITS TO LOW
0271 06030 001200 RAL
0272 06031 072176R STA CLIST
0273 06032 060001 LDA B
0274 06033 010073 AND M177 MASK LOW 7 BITS
0275 06034 066176R LDB CLIST
0276 06035 126024R JMP RRL9,I
0277* CLRLT CLEARS THE CURRENT BP LINKAGE ADDRESSES IN LST
0278* TO MAKE THEM UNAVAILABLE TO THE SUCCEEDING PROGRAMS. THIS CONSIST
0279* OF CLEARING WORDS OF THE LST ENTRY AND THE CORRESPONDING BP LINK
0280* VALUE IN THE BASE PAGE BUFFER.
0281* CALLING SEQUENCE:
0282* LDA <LOW BASE PAGE ADDRESS>
0283* LDB <HIGH BASE PAGE ADDRESS>
0284* JSB CLRLT
0285 06036 000000 CLRLT NOP
0286 06037 003004 CMA,INA
0287 06040 073101R STA LOWER SAVE LOWER BOUND
0288 06041 007004 CMB,INB
0289 06042 077145R STB UPPER SAVE UPPER BOUND
0290 06043 016000R JSB INLST INITIALIZE LST SCAN
0291 06044 016004R CLRNX JSB LSTX SET LST ADDRESSES
0292 06045 026061R JMP CLBPS GO TO CLEAR BP
0293 06046 163001R LDA LST5,I CHECK IF BP LINK TO BE CLEARED

```

```

0294 06047 043101R ADA LOWER
0295 06050 002020 SSA
0296 06051 026044R JMP CLRNX TRY NEXT ENTRY
0297 06052 163001R LDA LST5,I
0298 06053 043145R ADA UPPER
0299 06054 002021 SSA,RSS
0300 06055 026044R JMP CLRNX TRY NEXT ENTRY
0301 06056 006400 CLB
0302 06057 177001R STB LST5,I CLEAR BP LINK ADDRESS
0303 06060 026044R JMP CLRNX TRY NEXT ENTRY
0304 06061 067101R CLBPS LDB LOWER CLEAR BP LINK BUFFER BETWEEN
0305 06062 007004 CMB,INB UPPR AND LOWER LIMITS
0306 06063 063145R LDA UPPER
0307 06064 040001 ADA B
0308 06065 002021 SSA,RSS
0309 06066 126036R JMP CLRLT,I RETURN
0310 06067 073150R STA WDCNT
0311 06070 047205R ADB BPBUF ADJUST FOR BASE PAGE
0312 06071 047026R ADB BPOIT
0313 06072 002400 CLA
0314 06073 170001 STA B,I
0315 06074 006004 INB
0316 06075 037150R ISZ WDCNT
0317 06076 026073R JMP *-3
0318 06077 126036R JMP CLRLT,I RETURN
0319* BPOUT OUTPUTS THE BASE PAGE SECTION OF CODE FOLLOWING THE
0320* LOADING OF EACH MAIN OR SEGMENT. DISC ADDRESS IN DSKAD.
0321* CALLING SEQUENCE:
0322* LDA <UPPER BP ADDRESS>
0323* LDB <LOWER BP ADDRESS>
0324* JSB BPOUT
0325 06100 000000 BPOUT NOP
0326 06101 003004 CMA,INA
0327 06102 040001 ADA B
0328 06103 073136R STA TCNT SAVE BP LENGHT
0329 06104 047205R ADB BPBUF ADJUST FOR BP ADDRESS
0330 06105 047026R ADB BPOIT
0331 06106 077037R STB CURAT SAVE CURRENT LOW CORE ADDRESS
0332 06107 002021 SSA,RSS
0333 06110 126100R JMP BPOUT,I RETURN=ALL CODE OUT
0334 06111 017537R BPSYO JSB SETCR USER AREA
0335 06112 063050R LDA DSKAD CURRENT DISC ADDRESS
0336 06113 067037R LDB CURAT CURRENT LOW CORE ADDRESS
0337 06114 016243R JSB DISKO OUTPUT CURRENT BP SECTOR
0338 06115 060174 LDA TSONE CURRENT DISC ADDRESS
0339 06116 073050R STA DSKAD
0340 06117 067136R LDB TCNT
0341 06120 047273R ADB .128
0342 06121 077136R STB TCNT SAVE REMAINING COUNT
0343 06122 006021 SSB,RSS CHECK IF MORE TO OUTPUT
0344 06123 126100R JMP BPOUT,I RETURN=ALL CODE OUT
0345 06124 067037R LDB CURAT
0346 06125 047273R ADB .128
0347 06126 077037R STB CURAT SET NEXT CORE ADDRESS
0348 06127 026111R JMP BPSYO OUTPUT NEXT BP SECTOR TO DISC
0349* CONVD CONVERTS CONTENTS OF A TO ASCII DECIMAL OR OCTAL AND

```

```

0350*   STORES RESULT AT LOCATION SPECIFIED IN B.
0351*   CALLING SEQUENCE:
0352*       LDA <# TO BE CONVERTED> +OCTAL; -DECIMAL
0353*       LDB <ADDR TO STORE RESULT>
0354*       JSB CONVD
0355   06130 000000 CONVD NOP
0356   06131 077037R   STB CURAT   SET STORE ADDRESS
0357   06132 067305R   LDB OPWRS   ADDRESS OF OCTAL LOWERS
0358   06133 002020   SSA
0359   06134 067313R   LDB DPWRS   ADDRESS OF DECIMAL POWERS
0360   06135 077122R   STB RANAD   SET POWER RANGE ADDRESS
0361   06136 002021   SSA,RSS
0362   06137 003004   CMA,INA   CONVERT # TO NEGATIVE
0363   06140 070001   STA B
0364   06141 060051   LDA N2
0365   06142 073136R   STA TCNT   SET CONVERSION COUNTER
0366   06143 016161R   JSB GETD
0367   06144 033211R   IOR UBLNK
0368   06145 173037R   STA CURAT,I  SAVE BLANK,FIRST DIGIT
0369   06146 037037R   ISZ CURAT
0370   06147 016161R NEXTD JSB GETD   GET NEXT DIGIT
0371   06150 001727   ALF,ALF
0372   06151 173037R   STA CURAT,I  SAVE AS UPPER CHARACTER
0373   06152 016161R   JSB GETD   GET NEXT DIGIT
0374   06153 133037R   IOR CURAT,I  ADD UPPER CHARACTER
0375   06154 173037R   STA CURAT,I  STORE NEXT 2 CHARACTERS
0376   06155 037037R   ISZ CURAT
0377   06156 037136R   ISZ TCNT
0378   06157 026147R   JMP NEXTD   GET NEXT 2 DIGITS
0379   06160 126130R   JMP CONVD,I  RETURN -DONE
0380   06161 000000 GETD  NOP
0381   06162 002400   CLA
0382   06163 147122R INCRA ADB RANAD,I  ADD POWER
0383   06164 007026   CMB,SSB,INB,SZB
0384   06165 026171R   JMP ++4   DIGIT FOUND
0385   06166 002004   INA   INCR DIGIT
0386   06167 007004   CMB,INB   RESTORE REMAINDER TO NEG
0387   06170 026163R   JMP INCRA   RRY NEXT HIGHER DIGIT
0388   06171 147122R   ADB RANAD,I  ADD POWER
0389   06172 007004   CMB,INB   RESTORE REMAINDER
0390   06173 037122R   ISZ RANAD   INCR POWER LIST ADDRESS
0391   06174 033264R   IOR M60   CONVERT TO ASCII
0392   06175 126161R   JMP GETD,I  RETURN
0393*
0394*   CLIST CLEARS MEMORY MAP BUFFER WITH BLANKS
0395*
0396   06176 000000 CLIST NOP
0397   06177 067300R   LDB AMLST   ADDRESS OF MLIST
0398   06200 060043   LDA N8
0399   06201 073016R   STA AMAD
0400   06202 063207R   LDA BLNKS   TWO ASCII BLANKS
0401   06203 170001   STA B,I   CLEAR BUFFER WORD
0402   06204 006004   INB
0403   06205 037016R   ISZ AMAD
0404   06206 026203R   JMP *-3
0405   06207 126176R   JMP CLIST,I  RETURN-BUFFER CLEARED

```

```

0406*
0407*      IDSCN SCANS PROGRAM IDENT LIST FOR PROGRAM MAIN OR SEGMENT.
0408*      PROGRAM TYPE SET IN PTYPE.
0409*      CALLING SEQUENCE:
0410*          JSB IDSCN
0411*      RETURN
0412*          (N+1) = END OF IDENT8
0413*          (N+2) = PROG FOUND E=M/S FOR CURRENT PROGRAM.
0414*
0415      06210 000000 IDSCN NOP
0416      06211 063005R LDA CIDNT GET NEXT IDENT IN SCAN
0417      06212 073003R STA TIDNT SET IDENT ADDRESS FOR IDX
0418      06213 016035R JSB IDX SET IDENT ADDRESS
0419      06214 126210R JMP IDSCN,I RETURN-END OF IDENT8
0420      06215 063006R LDA ID1 GET CURRENT MAIN IDENT ADDRESS
0421      06216 073074R STA IMAIN
0422      06217 063003R LDA TIDNT
0423      06220 073005R STA CIDNT SAVE NEXT IDENT ADDR
0424      06221 163013R LDA ID6,I GET TYPE
0425      06222 001265 RAL,CLE,ERA SET E=M/S
0426      06223 010073 AND M177
0427      06224 053121R CPA PTYPE CURRENT PROGRAM TYPE?
0428      06225 002001 RSS YES
0429      06226 026213R JMP IDSCN+3 NO-GO TRY NEXT IDENT
0430      06227 036210R ISZ IDSCN INCR RETURN ADDRESS
0431      06230 126210R JMP IDSCN,I RETURN
0432*
0433*      GETP DECIPHERS OPERATOR PARAMETER INPUT DATA.
0434*      RESULTS STORED IN PBUF AND A REGISTER INDICATES TERMINATION
0435*      CHARACTER; 0 INDICATES END OF INPUT, COMMA, LEFT PARENTHESIS, AND
0436*      RIGHT PARENTHESIS INDICATES THAT MORE DATA FOLLOWS.
0437*
0438      06231 000000 GETP NOP
0439      06232 063207R LDA BLNKS SET PBUF TO ASCII BLANKS
0440      06233 072214R STA PBUF
0441      06234 072215R STA PBUF+1
0442      06235 072216R STA PBUF+2
0443      06236 062365R LDA NEGPM
0444      06237 002003 SZA,RSS
0445      06240 126231R JMP GETP,I
0446      06241 063303R LDA DFPBF SET CONVERTED CHARACTER STORE
0447      06242 000066 CLE,ELA ADDRESS
0448      06243 073151R STA XPREL
0449      06244 066366R GET10 LDB PMBUF
0450      06245 004065 CLE,ERB CONVERT TO WORD ADDRESS
0451      06246 160001 LDA B,I GET DATA WORD FROM INPUT BUFFER
0452      06247 002041 SEZ,RSS CHECK IF HI OR LOW CHAR
0453      06250 001727 ALF,ALF SHIFT HI INTO LOW
0454      06251 010074 AND M377 ISOLATE CHARACTER
0455      06252 005600 ELB INCREMENT INPUT BUFFER CHARACTER
0456      06253 006004 INB ADDRESS
0457      06254 076366R STB PMBUF
0458      06255 053261R CPA COMMA COMMA?
0459      06256 026312R JMP GET20
0460      06257 053262R CPA LPARN LEFT PARENTHESIS?
0461      06260 026312R JMP GET20

```

0462	06261	053263R	CPA RPARN	RIGHT PARENTHESIS?
0463	06262	026312R	JMP GET20	
0464	06263	052315R	CPA SLASH	
0465	06264	026311R	JMP GET15	
0466	06265	067151R	LDB XPREL	GET STORE CHARACTER ADDRESS
0467	06266	004065	CLE,ERB	CONVERT TO WORD ADDRESS
0468	06267	002041	SEZ,RSS	CHECK IF HI OR LOW STORE POSIT.
0469	06270	001727	ALF,ALF	SHIFT INTO HIGH
0470	06271	073065R	STA EXCNT	SAVE CHARACTER
0471	06272	160001	LDA B,I	GET STORED BUFFER WORD
0472	06273	002040	SEZ	
0473	06274	026277R	JMP ++3	
0474	06275	010074	AND M377	SAVE LOW CHARACTER
0475	06276	002001	RSS	
0476	06277	010075	AND M1774	SAVE HI CHARACTER
0477	06300	033065R	IOR EXCNT	IOR IN CURRENT CHARACTER
0478	06301	170001	STA B,I	SAVE IN BUFFER
0479	06302	005600	ELB	
0480	06303	006004	INB	INCREMENT STORE ADDRESS
0481	06304	077151R	STB XPREL	
0482	06305	036365R	ISZ NEGPM	
0483	06306	026244R	JMP GET10	GO TO PROCESS NEXT CHARACTER
0484	06307	002400	CLA	SET A TO INDICATE END
0485	06310	126231R	JMP GETP,I	RETURN
0486	06311	063212R	GET15 LDA SLSHE	
0487	06312	036365R	GET20 ISZ NEGPM	
0488	06313	126231R	JMP GETP,I	
0489	06314	126231R	JMP GETP,I	
0490	06315	000057	SLASH OCT 57	
0491*				
0492*				
0493*				
0494*				
0495*				
0496*				
0497	06316	000000	PROGL NOP	
0498	06317	002400	CLA	CLEAR PROGRAM NAME COUNTER
0499	06320	073110R	STA PCNTT	
0500	06321	063304R	LDA PLAD	SET PROGRAM NAME BUFFER ADDRESS
0501	06322	072364R	STA PCONT	
0502	06323	063141R	LDA TERMN	CHECK IF AT END OF PARAMETERS
0503	06324	002003	SZA,RSS	
0504	06325	126316R	JMP PROGL,I	END=80 RETURN
0505	06326	053262R	CPA LPARN	CHECK IF LEFT PARENTHESIS
0506	06327	002001	RSS	YES=80 PROGRAMS SPECIFIED
0507	06330	126316R	JMP PROGL,I	NO=80 RETURN
0508	06331	016231R	PROG5 JSB GETP	GET ONE CONTROL PARAMETER
0509	06332	073141R	STA TERMN	
0510	06333	002003	SZA,RSS	
0511	06334	126316R	JMP PROGL,I	
0512	06335	062214R	LDA PBUF	
0513	06336	172364R	STA PCONT,I	SAVE PROG NAME IN BUFFER
0514	06337	036364R	ISZ PCONT	
0515	06340	062215R	LDA PBUF+1	
0516	06341	172364R	STA PCONT,I	
0517	06342	036364R	ISZ PCONT	

0518	06343	062216R	LDA	PBUF+2	
0519	06344	010075	AND	M1774	
0520	06345	172364R	STA	PCONT,I	
0521	06346	036364R	ISZ	PCONT	
0522	06347	037110R	ISZ	PCNTT	INCREMENT PROGRAM NAME COUNTER
0523	06350	063141R	LDA	TERMN	
0524	06351	053263R	CPA	RPARN	CHECK IF AT END OF PROG LIST
0525	06352	002001	RSS		YES, SO CHECK IF MORE FOLLOWS
0526	06353	026361R	JMP	PROG9	GO CHECK IF AT END OF INPUT
0527	06354	002003	SZA,RSS		
0528	06355	126316R	JMP	PROGL,I	RETURN=NO MORE INPUT
0529	06356	016231R	JSB	GETP	SKIP OVER NEXT CHARACTER TO GET
0530	06357	073141R	STA	TERMN	TRUE TERMINATION SYMBOL
0531	06360	126316R	JMP	PROGL,I	RETURN FOR PROGRAM PROCESSING
0532	06361	002002	PROG9	SZA	CHECK IF AT END OF INPUT
0533	06362	026331R	JMP	PROG5	NO-GET NEXT PARAMETER
0534	06363	126316R	JMP	PROGL,I	YES=SO RETURN
0535	06364	000000	PCONT	NOP	
0536	06365	000000	NEGPM	NOP	NEG # INPUT CHAR COUNT
0537	06366	000000	PMBUF	NOP	INPUT BUFFER CHAR ADDR

0539*

0540* ERROR MESSAGES

0541*

0542 06367 063167R LOVER LDA ERR05

OUTPUT L05 MESSAGE

0543 06370 016114R ABRTX JSB ERROR

0544 06371 063173R ABORT LDA .16

OUTPUT: LOADR TERMINATED MESSAGE

0545 06372 066662R LOB MESAB

0546 06373 016101R JSB SYOUT

0547 06374 026614R JMP LTERM

0548*

0549* SET SIMULATED END-OF-TAPE

0550*

0551 06375 000000 SET NOP

0552 06376 063112R LDA PGMIN

0553 06377 010072 AND M77

0554 06400 002003 SZA,RSS

0555 06401 126375R JMP SET,I

0556 06402 050055 CPA .2

0557 06403 126375R JMP SET,I

0558 06404 033177R IOR M700

0559 06405 073140R STA TEMP

0560 06406 016001X JSB EXEC

SET SIMULATED END OF TAPE

0561 06407 006412R DEF **3

0562 06410 000056 DEF .3

0563 06411 011140R DEF TEMP

0564 06412 126375R JMP SET,I

0566*
 0567* THE FOLLOWING ROUTINES PROCESS TWO RECORDS ENTERED
 0568* VIA THE BATCH DEVICE. BOTH RECORDS SHOULD CONTAIN
 0569* TWO OCTAL NUMBERS (MAX 5 DIGITS EACH) SEPARATED
 0570* BY EITHER A BLANK OR A COMMA. IF EITHER OF THESE
 0571* NUMBERS IS OMITTED, THE DEFAULT SYSTEM VALUE WILL
 0572* BE ASSUMED. THE FIRST RECORD IS INTERPRETED AS
 0573* LOWER & UPPER BASE PAGE BOUNDS. THE SECOND
 0574* RECORD IS INTERPRETED AS LOWER & UPPER MAIN
 0575* PROGRAM BOUNDS.

0576*
 0577* IF THE BATCH DEVICE IS THE SYSTEM CONSOLE, INPUT
 0578* OF THE RECORDS WILL BE PROMPTED BY:

0579*
 0580* BP BND (L,U)?
 0581* PROG BND (L,U)?

0582*
 0583* AN ERROR IN THE FIRST RECORD WILL RESULT IN AN
 0584* L18 ERROR; AN ERROR IN THE SECOND AN L19. FOL-
 0585* LOWING THE ERROR, THE RECORD MAY BE REENTERED.

0586*
 0587* IF THE BATCH DEVICE IS NOT THE SYSTEM CONSOLE,
 0588* INPUT OF THE RECORDS SHOULD IMMEDIATELY FOLLOW
 0589* THE ;PR,LOADR DIRECTIVE. ANY ERROR IN EITHER WILL
 0590* CAUSE THE LOADER TO ABORT.

0591*

0592	06413	062454R	STBND	LDA BMES	SET FOR
0593	06414	006400		CLB	BASE PAGE BOUNDS
0594	06415	016476R		JSB STB	ENTER & PROCESS BOUNDS
0595	06416	026434R		JMP BPBE	ERROR RETURN
0596	06417	073152R		STA UBFWA	"A" = LOWER BP BOUND
0597	06420	077153R		STB UBLWA	"B" = UPPER BP BOUND
0598	06421	007004		CMB, INB	MAKE SURE BOUNDS ARE
0599	06422	046453R		ADB B1777	ON THE BASE PAGE
0600	06423	006020		SSB	(< 2000B)
0601	06424	026434R		JMP BPBE	NO = ERROR

0602*

0603	06425	062465R	STBI	LDA BMES1	SET FOR
0604	06426	006404		CLB, INB	MAIN PROGRAM BOUNDS
0605	06427	016476R		JSB STB	ENTER & PROCESS BOUNDS
0606	06430	026442R		JMP PBE	ERROR RETURN
0607	06431	073154R		STA UMFWA	"A" = LOWER MAIN PROG BOUND
0608	06432	077155R		STB UMLWA	"B" = UPPER MAIN PROG BOUND
0609	06433	026034R		JMP LOAD1	RETURN TO MAIN PROCESSING

0611* ERROR ROUTINES

0613	06434	062451R	BPBE	LDA ERR18	BASE PAGE
0614	06435	016114R		JSB ERROR	BOUND ERROR = L18
0615	06436	060111		LDA BATCH	INPUT FROM
0616	06437	050112		CPA SYSTY	SYSTEM CONSOLE?
0617	06440	026413R		JMP STBND	YES = REENTER
0618	06441	026371R		JMP ABORT	NO = ABORT

0620	06442	062452R	PBE	LDA ERR19	MAIN PROGRAM
------	-------	---------	-----	-----------	--------------

PAGE 0086 #06 DOS-III LOADER - DISC I/O ROUTINES

0621	06443	016114R	JSB ERROR	BOUND ERROR = L19
0622	06444	060111	LDA BATCH	INPUT FROM
0623	06445	050112	CPA SYSTY	SYSTEM CONSOLE?
0624	06446	026425R	JMP STB1	YES = REENTER
0625	06447	026371R	JMP ABORT	NO = ABORT

0627* CONSTANTS, ETC. FOR PROGRAM BOUNDS ROUTINES

0629	06450	177720	NB60	OCT	=60
0630	06451	030470	ERR18	ASC	1,18
0631	06452	030471	ERR19	ASC	1,19
0632	06453	001777	B1777	OCT	1777
0633	06454	006455R	BMES	DEF	++1
0634	06455	041120		ASC	8,BP BND (L,U)?
0635	06455	006466R	BMES1	DEF	++1
0636	06466	050122		ASC	8,PROG BND (L,U)?

0638* ROUTINE TO ENTER TWO ASCII NUMBERS FROM THE BATCH
 0639* DEVICE AND CONVERT THEM TO OCTAL.

0640*

0641* UPON ENTRY:

0642*

0643* "A" = BUFFER ADDRESS OF MESSAGE TO BE
 0644* OUTPUT IF BATCH IS SYSTEM CONSOLE

0645* "B" = BASE PAGE/MAIN PROGRAM FLAG

0646* 0 => BASE PAGE

0647* 1 => MAIN PROGRAM

0648*

0649* ON EXIT:

0650*

0651* P+1 : ERROR RETURN

0652* ILLEGAL OCTAL DIGIT

0653* LOWER > UPPER BOUND

0654* NUMBER > 8 DIGITS

0655*

0656* P+2 : NORMAL RETURN

0657* "A" = LOWER BOUND

0658* "B" = UPPER BOUND

0659*

0660* PARAMETERS USED:

0661*

0662* DATAB = INPUT BUFFER

0663* PLIST = 3-WORD ASCII BUFFER FOR CONV

0664* ID1 = POINTER INTO PLIST

0665* ID2 = SUM FOR OCTAL NUMBER CONVERSION

0666* ID3 = BASE PAGE/MAIN PROG FLAG

0667* ID4 = LOWER BOUND

0668* ID5 = LOWER/UPPER BOUND FLAG

0669* ID6 = POINTER INTO DATAB

0670* ID7 = UPPER/LOWER FLAG FOR DATAB

0001* LST1 = UPPER/LOWER FLAG FOR PLIST

0002* LST2 = IGNORE BLANKS FLAG

0003*

0004 06476 000000 STB NOP

0005 06477 072515R STA BBADD STORE MESSAGE BUFFER ADDRESS

0006 06500 077010R STB ID3 SAVE BP/MAIN PROG FLAG

0007*

0008 06501 060111 STB2 LDA BATCH IF BATCH IS

0009 06502 050112 CPA SYSTY SYSTEM CONSOLE, SET

0010 06503 033267R IOR M400 ECHO BIT IN

0011 06504 073140R STA TEMP CONTROL WORD

0012 06505 060111 LDA BATCH IF BATCH IS

0013 06506 050112 CPA SYSTY NOT SYSTEM CONSOLE,

0014 06507 002001 RSS SKIP OUTPUTTING OF

0015 06510 026517R JMP STB3 MESSAGE

0016*

0017 06511 016001X JSB EXEC OUTPUT MESSAGE TO SYSTEM CONSOLE

0018 06512 006517R DEF ++5

0019 06513 000055 DEF .2

0020 06514 000054 DEF .1

0021 06515 000000 BBADD NOP BUFFER ADDRESS STORED HERE

0022 06516 000063 DEF .8

0023*

0024	06517	016001X	STB3	JSB EXEC	INPUT TWO ASCII NUMBERS
0025	06520	006525R		DEF **5	INTO DATAB
0026	06521	000054		DEF ,1	
0027	06522	011140R		DEF TEMP	
0028	06523	010726R		DEF DATAB	
0029	06524	007166R		DEF N72	
0030	06525	006003		SZB,RSS	
0031	06526	026501R		JMP STB2	TRY AGAIN IF NO CHAR INPUT
0032	06527	007004		CMB,INB	
0033	06530	076365R		STB NEGPM	SAVE ** CHARS INPUT
0034*					
0035	06531	002400		CLA	SET FOR
0036	06532	073012R		STA ID5	LOWER BOUND
0037	06533	062273R		LDA DATAB	START SCAN AT BEGINNING
0038	06534	073013R		STA ID6	OF DATAB
0039	06535	003400		CCA	SET UPPER/LOWER FLAG
0040	06536	073014R		STA ID7	TO UPPER
0041*					
0042	06537	002400	STB4	CLA	CLEAR
0043	06540	072000R		STA PLIST	ASCII BUFFER
0044	06541	072001R		STA PLIST+1	
0045	06542	072002R		STA PLIST+2	
0046	06543	063304R		LDA PLAD	INITIALIZE ASCII
0047	06544	073006R		STA ID1	BUFFER POINTER
0048	06545	003400		CCA	SET UPPER/LOWER FLAG
0049	06546	072775R		STA LST1	TO UPPER
0050	06547	072776R		STA LST2	SET TO IGNORE BLANKS
0051*					
0052	06550	067014R	STB5	LDB ID7	GET CHARACTER
0053	06551	163013R		LDA ID6,I	
0054	06552	006002		SZB	
0055	06553	001727		ALF,ALF	
0056	06554	010073		AND M177	
0057	06555	007002		CMB,SZB	
0058	06556	037013R		ISZ ID6	
0059	06557	077014R		STB ID7	SET NEW UPPER/LOWER FLAG
0060*					
0061	06560	053210R		CPA BLANK	IS IT A BLANK?
0062	06561	026620R		JMP PRBLN	YES - PROCESS BLANK
0063	06562	053261R		CPA COMMA	IS IT A COMMA?
0064	06563	026614R		JMP PRCMA	YES - PROCESS COMMA
0065*					
0066	06564	066450R		LDB NB60	MAKE SURE VALID OCTAL DIGIT
0067	06565	044000		ADB A	
0068	06566	006020		SSB	
0069	06567	126476R		JMP STB,I	<60 ERROR
0070	06570	044043		ADB N8	
0071	06571	006021		SSB,RSS	
0072	06572	126476R		JMP STB,I	>67 ERROR
0073	06573	072776R		STA LST2	<ST2>0, SO DON'T IGNORE BLANKS
0074*					
0075	06574	066775R		LDB LST1	STORE CHARACTER IN ASCII BUFFER
0076	06575	006002		SZB	
0077	06576	001727		ALF,ALF	UPPER CHARACTER
0078	06577	133006R		IOR ID1,I	
0079	06600	173006R		STA ID1,I	

```

0080 06601 006002      SZB
0081 06602 026607R    JMP SCHK
0082 06603 063304R    LDA PLAD
0083 06604 040055      ADA .2
0084 06605 053006R    CPA ID1
0085 06606 126476R    JMP STB,I      6 DIGITS - ERROR
0086 06607 007002 SCHK CMB,SZB
0087 06610 037006R    ISZ ID1
0088 06611 076775R    STB LST1
0089*
0090 06612 016627R    JSB DONE      DONE?
0091 06613 026550R    JMP STB5      NO- GET NEXT CHARACTER
0092*
0093 06614 003400 PRMA CCA      SET TO
0094 06615 072776R    STA LST2      IGNORE BLANKS
0095 06616 016627R    JSB DONE      DONE?
0096 06617 026625R    JMP PRN      NO- CONVERT # & CONTINUE
0097*
0098 06620 062776R PRBLN LDA LST2  IGNORE
0099 06621 002021      SSA,RSS      BLANKS ?
0100 06622 026625R    JMP PRN      NO - CONVERT #
0101 06623 016627R    JSB DONE      YES- DONE?
0102 06624 026550R    JMP STB5      NO- GET NEXT CHAR
0103*
0104 06625 016640R PRN JSB CONV    CONVERT ASCII TO OCTAL
0105 06626 026537R    JMP STB4      RETURN IF 2ND NUMBER NEEDED

```

0107* ROUTINE TO TEST WHETHER OR NOT ALL CHARACTERS

0108* INPUT HAVE BEEN PROCESSED

0109*

0110* RETURN: P+1 - MORE CHARACTERS TO PROCESS

0111* NO RETURN IF ALL CHARACTERS HAVE BEEN PROCESSED.

0112* NUMBER IN ASCII BUFFER WILL BE CONVERTED.

0113* IF ONLY ONE NUMBER HAS BEEN ENTERED,

0114* 2ND NUMBER WILL BE DEFAULT VALUE.

0115*

0116 06627 000000 DONE NOP

0117 06630 036365R ISZ NEGPM

MORE CHARACTERS?

0118 06631 126627R JMP DONE,I

YES- CONTINUE

0119 06632 016640R DONE1 JSB CONV

CONVERT # IN ASCII BUFFER

0120 06633 002400 CLA

RETURN IF 2ND NUMBER NEEDED

0121 06634 072000R STA PLIST

CLEAR BUFFER FOR

0122 06635 072001R STA PLIST+1

DEFAULT VALUE

0123 06636 072002R STA PLIST+2

0124 06637 026632R JMP DONE1

CONVERT 2ND NUMBER

0126* ROUTINE TO CONVERT NUMBER IN ASCII BUFFER TO OCTAL.

0127* INPUT IS TERMINATED BY A NULL. IF VALUE=0,

0128* THE SYSTEM DEFAULT VALUE IS ASSUMED.

0129* THE FIRST TIME THIS ROUTINE IS CALLED, IT CONVERTS

0130* THE LOWER BOUND, SAVES IT AND RETURNS. THE SECOND

0131* TIME, IT CONVERTS THE UPPER BOUND, THEN SETS

```

0132*
0133*      "A" = LOWER BOUND
0134*      "B" = UPPER BOUND
0135* AND RETURNS TO P+2 OF STB ROUTINE.
0136*
0137 06640 000000 CONV NOP
0138 06641 063304R LDA PLAD INITIALIZE ASCII BUFFER
0139 06642 073006R STA ID1 POINTER
0140 06643 006400 CLB INITIALIZE OCTAL
0141 06644 077007R STB ID2 SUM TO ZERO
0142*
0143 06645 163006R CONV1 LDA ID1,I GET 1ST CHAR
0144 06646 001727 ALF,ALF
0145 06647 010073 AND M177
0146 06650 016656R JSB CON CONVERT IT
0147 06651 163006R LDA ID1,I GET 2ND CHAR
0148 06652 010073 AND M177
0149 06653 016656R JSB CON CONVERT IT
0150 06654 037006R ISZ ID1 SET FOR NEXT WORD
0151 06655 026645R JMP CONV1

0153 06656 000000 CON NOP
0154 06657 002003 SZA,RSS IS CHAR NULL?
0155 06660 026670R JMP CONV2 YES - INPUT TERMINATED
0156 06661 010062 AND ,7 N= MASK OFF 60B
0157*
0158 06662 067007R LDB ID2 UPDATE OCTAL SUM
0159 06663 005222 RBL,RBL
0160 06664 005200 RBL
0161 06665 044000 ADB A
0162 06666 077007R STB ID2
0163 06667 126656R JMP CON,I
0164*
0165 06670 006002 CONV2 SZB DEFAULT BOUNDS NECESSARY?
0166 06671 026720R JMP CONV3 NO - BET GIVEN BOUNDS
0167*
0168 06672 063012R LDA ID6 PROCESSING
0169 06673 002002 SZA LOWER BOUND?
0170 06674 026704R JMP UPPR NO - UPPER
0171 06675 064255 LDB SBFWA SET FOR BASE PAGE LOWER
0172 06676 063010R LDA ID3 IS IT FOR
0173 06677 002002 SZA BASE PAGE?
0174 06700 064254 LDB SMFWA NO - CHANGE TO MAIN PRG LOWER
0175 06701 077011R LWR STB ID4 SAVE LOWER BOUND
0176 06702 077012R STB ID5 SET FOR UPPER BOUND NEXT
0177 06703 126640R JMP CONV,I RETURN
0178*
0179 06704 064256 UPPR LDB SBLWA SET FOR BASE PAGE UPPER
0180 06705 063010R LDA ID3 IS IT FOR BASEE PAGE?
0181 06706 002002 SZA
0182 06707 064100 LDB SMLWA NO - CHANGE TO MAIN PROG UPPE
0183*
0184 06710 063011R UPPR1 LDA ID4 MAKE SURE LOWER <= UPPER
0185 06711 003004 CMA,INA
0186 06712 040001 ADA B

```

0187	06713	002020	SSA	
0188	06714	126476R	JMP STB,I	NO - ERROR RETURN
0189*				
0190	06715	063011R	LDA ID4	"A" = LOWER BOUND
0191	06716	036476R	ISZ STB	"B" = UPPER BOUND
0192	06717	126476R	JMP STB,I	RETURN TO P+2 OF STB
0193*				
0194	06720	063012R	CONV3 LDA ID5	IS THIS FOR LOWER BOUND?
0195	06721	002002	SZA	
0196	06722	026710R	JMP UPPR1	NO
0197	06723	026701R	JMP LWR	YES

0199*
0200* THE FOLLOWING ROUTINES LINK A PROGRAM THROUGH
0201* CURRENT PAGE LINKS.
0202*
0203* ALINK - CALLED AFTER NAM RECORD IS READ.
0204* DETERMINES WHETHER OR NOT CURRENT
0205* PAGE LINKS ARE TO BE USED. SETS UP
0206* LOW LINK AREA (BOTH INITIAL GUESS &
0207* FIXED LENGTH FOLLOWING PASS 1)
0208*
0209* BLINK - CALLED AFTER END RECORD IS READ.
0210* IF UPPER LINK AREA IS PRESENT, IT IS
0211* OUTPUT TO THE DISC.
0212*
0213* DLINK - CALLED FROM BPSCN AFTER FAILING TO
0214* FIND DESIRED LINK IN BASE PAGE, BUT
0215* PRIOR TO ALLOCATING A NEW BASE PAGE
0216* LINK. IF NO CURRENT PAGE LINK IS POS-
0217* SIBLE, RETURN IS MADE TO P+1 & A BP
0218* LINK IS ALLOCATED. IF A CURRENT PAGE
0219* LINK IS POSSIBLE, IT IS CREATED &
0220* RETURN IS MADE TO LINKC+1.
0221*
0222* CROSS - CALLED FROM DLINK WHEN A PAGE
0223* BOUNDARY IS CROSSED OR FROM END
0224* RECORD DETECTED BEFORE PASS 3. FIRST
0225* PAGE CROSSING SETS FOR PASS 2 & FIXES
0226* LENGTH OF LOW LINK AREA. SECOND PAGE
0227* CROSSING SETS FOR PASS 3 & OUTPUTS
0228* LOW LINK AREA TO DISC. THIRD PAGE
0229* CROSSING RESULTS IN DEFAULTING LINK
0230* TO BASE PAGE.
0231*

0233	06724	000000	ALINK	NOP	
0234	06725	063157R	LDA	L01	CURRENT PAGE
0235	06726	002020	SSA		LINKS POSSIBLE?
0236	06727	126724R	JMP	ALINK,I	NO - RETURN
0237	06730	002002	SZA		YES - PASS 3?
0238	06731	027003R	JMP	PS3	YES - SET LOW LINK POINTER
0240	06732	063156R	LDA	L001	PASS 1 - FOR LOW LINK
0241	06733	002002	SZA		AREA GUESS?
0242	06734	026773R	JMP	PS2	NO - PASS 2 FOR FIXED LOW LNK
0244	06735	063155R	LDA	CPL	CURRENT PAGE LINKS REQUESTED
0245	06736	002003	SZA,RSS		IN CALLING SEQUENCE?
0246	06737	026771R	JMP	NCPL	NO - SET L01 < 0 FOR NO CPL
0248	06740	063323R	LDA	TPREL	GET ADDRESS
0249	06741	070001	STA	B	OF LAST WORD
0250	06742	033201R	IOR	01777	OF PAGE
0251*					
0252	06743	007004	CMB,INB		COMPUTE #
0253	06744	006004	INB		WORDS

0254	06745	044000		ADB A	REMAINING
0255	06746	077150R		STB TEMP2	ON PAGE
0256*					
0257	06747	063115R		LDA PLGTH	COMPUTE # WORDS
0258	06750	001265		RAL,CLE,ERA	OF PROGRAM
0259	06751	007004		CMB,INB	THAT FALL
0260	06752	044000		ADB A	BEYOND THIS
0261	06753	077147R		STB TEMP1	PAGE
0262*					
0263	06754	006020		SSB	IF PROGRAM FITS ON ONE PAGE
0264	06755	026771R		JMP NCPL	CURRENT PAGE LINKS ARE NOT
0265	06756	006003		SZB,RSS	REQUIRED.
0266	06757	026771R		JMP NCPL	
0267*					
0268	06760	063150R		LDA TEMP2	COMPUTE MINIMUM OF:
0269	06761	001100		ARS	HALF # WORDS OF PROGRAM
0270	06762	007004		CMB,INB	ON CURRENT PAGE -OR-
0271	06763	044000		ADB A	# WORDS OF PROGRAM
0272	06764	006021		SSB,RSS	ON NEXT PAGE
0273*					
0274	06765	063147R		LDA TEMP1	DIVIDE THIS
0275	06766	001121		ARS,ARS	MINIMUM BY FOUR
0276	06767	002002		SZA	IF NON-ZERO,USE AS SIZE OF LOW
0277	06770	026774R		JMP PS1	LINK AREA. IF EQUAL TO ZERO,
0279	06771	003400	NCPL	CCA	SET L01 < 0 TO INDICATE
0280	06772	026747R		JMP LOADX	NO CURRENT PAGE LINKS
0282	06773	063152R	PS2	LDA LWH2	PASS 2 - FIXED LENGTH OF LOW LNK
0283	06774	067323R	PS1	LDB TPREL	SET NEW:
0284	06775	077147R		STB TEMP1	LOW LIMIT OF LOW LINK AREA
0285	06776	044000		ADB A	(ADD LENGTH OF LINK AREA)
0286	06777	077323R		STB TPREL	LOAD ADDRESS
0287	07000	077163R		STB CPMAX	UPPER LIMIT OF LOW LINK AREA
0288	07001	003004		CMA,INA	SAVE NEGATIVE LENGTH OF LINK
0289	07002	073152R		STA LWH2	AREA FOR OUTPUT TO DISC (BLIND)
0291	07003	063147R	PS3	LDA TEMP1	SET UP POINTERS
0292	07004	073162R		STA CPLOC	FOR LOW LINK AREA
0293	07005	073161R		STA CPBAS	
0294	07006	013203R		AND M0760	
0295	07007	073164R		STA CPADD	GET CURRENT PAGE ADDRESS
0296	07010	126724R		JMP ALINK,I	

0298	07011	000000	DLINK	NOP	
0299	07012	062343R		LDA BPSCN	WAS CALL FROM
0300	07013	053160R		CPA CLINK	DBLBPROCESSOR?
0301	07014	002001		RSS	
0302	07015	127011R		JMP DLINK,I	NO - USE BASE PAGE LINK
0303	07016	063167R		LDA L01	USING CURRENT
0304	07017	002020		SSA	PAGE LINKS?
0305	07020	127011R		JMP DLINK,I	NO - USE BASE PAGE LINK
0307	07021	063164R	SETL	LDA CPADD	GET LINK PAGE
0308	07022	053106R		CPA PAGNO	SAME AS INSTRUCTION PAGE?
0309	07023	002001		RSS	
0310	07024	027056R		JMP CROSS	NO - PAGE BOUNDARY CROSSING
0312	07025	062774R		LDA PLST	SET TO
0313	07026	073115R		STA BLINK	BOTTOM OF LINK AREA
0314	07027	067161R		LOB CPBAS	GET NUMBER OF CURRENT PAGE LINKS
0315	07030	007004		CMB,INB	ALLOCATED SO FAR. IF ZERO,
0316	07031	047162R		ADB CPLOC	JUMP TO ALLOCATE.
0317	07032	007007		CMB,INB,SZB,RSS	
0318	07033	027042R		JMP DLIN1	
0319	07034	163115R	DLIN2	LDA BLINK,I	IF NOT ZERO, SCAN CURRENT PAGE
0320	07035	053105R		CPA OPRND	LINKS PRESENT FOR POSS. MATCH
0321	07036	027042R		JMP DLIN1	MATCH FOUND- RETURN ADDRESS
0322	07037	037115R		ISZ BLINK	MATCH NOT FOUND- CONTINUE
0323	07040	006006		INB,SZB	SEARCHING THROUGH LINKS
0324	07041	027034R		JMP DLIN2	
0326	07042	063115R	DLIN1	LDA BLINK	MAKE SURE CORE BUFFER DOESN'T
0327	07043	053004R		CPA PIONT	OVERFLOW INTO ID TABLE
0328	07044	127011R		JMP DLINK,I	YES : USE BP LINKS
0329	07045	063162R		LDA CPLOC	GET NEXT AVAIL LINK LOC
0330	07046	040001		ADA B	ADD OFFSET IN CASE NOT NEW LINK
0331	07047	053163R		CPA CPMAX	OVERFLOW LINK AREA?
0332	07050	127011R		JMP DLINK,I	YES- MUST USE BP LINK
0333	07051	006003		SZB,RSS	IF NEW LINK, INCREMENT THE
0334	07052	037162R		ISZ CPLOC	LOCATION COUNTER
0335	07053	067105R		LOB OPRND	GET THE LINK WORD
0336	07054	177115R		STB BLINK,I	SET IN THE LINK AREA
0337	07055	027705R		JMP LINKC+1	RETURN TO DBL PROCESSING
0339	07056	063157R	CROSS	LDA L01	PASS 3?
0340	07057	002002		SZA	
0341	07060	027071R		JMP BLIN3	YES-ASSIGN UPPR LINK OR RETURN
0343	07061	063156R		LDA L001	PASS 1?
0344	07062	002003		SZA,RSS	
0345	07063	027107R		JMP FLLA	YES-FIX LOW LINK AREA
0347	07064	063152R		LDA LWH2	IF LENGTH OF LOW LINK AREA IS
0348	07065	002002		SZA	GREATER THAN ZERO,
0349	07066	017132R		JSB BLIN0	OUTPUT IT TO DISC
0350	07067	002404	BLIN2	CLA,INA	SET FOR PASS 3
0351	07070	026747R		JMP LOADX	AND GO

0353	07071	002001	BLIN3	RSS	SKIP THE 1ST TIME THROUGH, THEN
0354	07072	127011R		JMP DLINK,I	RSS CHANGED TO NOP - (BP LINKS)
0356	07073	063115R		LDA PLGTH	COMPUTE
0357	07074	001265		RAL,CLE,ERA	
0358	07075	043323R		ADA TPREL	
0359	07076	073162R		STA CPLOC	
0360	07077	073161R		STA CPBAS	LOW BOUND OF UPPER LINK AREA
0361	07100	013203R		AND M0760	
0362	07101	073164R		STA CPADD	CURRENT PAGE ADDRESS
0363	07102	043202R		ADA M2000	
0364	07103	073163R		STA CPMAX	UPPER BOUND OF UPPER LINK AREA
0365	07104	002400		CLA	CLEAR BLIN3, SO SUBSEQUENT PAGE
0366	07105	073071R		STA BLIN3	CROSSINGS WILL USE BP LINKS
0367	07106	027021R		JMP SETL	RETURN TO ALLOCATE CP LINK
0369	07107	063147R	FLLA	LDA TEMP1	COMPUTE # LINKS ACTUALLY USED IN
0370	07110	003004		CMA,INA	LOW LINK AREA, & SAVE FOR
0371	07111	043162R		ADA CPLOC	FIXED LENGTH.
0372	07112	073152R		STA LWH2	
0373	07113	002404		CLA,INA	SET FOR PASS 2
0374	07114	026732R		JMP LD3	AND GO
0376	07115	000000	BLINK	NOP	
0377	07116	037151R		ISZ LWH1	IF FROM END RECORD BEFORE PASS 3
0378	07117	127115R		JMP BLINK,I	NO UPPER LINK AREA, RETURN
0380	07120	063162R		LDA CPLOC	COMPUTE NEGATIVE #
0381	07121	003004		CMA,INA	OF WORDS IN UPPER LINK AREA.
0382	07122	043323R		ADA TPREL	(TPREL IS AT END OF CURRENT
0383	07123	073152R		STA LWH2	PROGRAM)
0384	07124	002003		SZA,RSS	IF LENGTH = 0, DON'T
0385	07125	127115R		JMP BLINK,I	OUTPUT
0386	07126	017132R		JSB BLIN0	OTHERWISE) OUTPUT UPPER LINK2
0388	07127	063162R		LDA CPLOC	ADJUST NEW RELOCATION ADDRESS TO
0389	07130	073323R		STA TPREL	BEYOND UPPER LINK AREA
0390	07131	127115R		JMP BLINK,I	
0392	07132	000000	BLIN0	NOP	
0393	07133	062774R		LDA PLST	SET POINTER TO START
0394	07134	072724R		STA ALINK	OF LINK BUFFER
0395	07135	067161R		LDB CPBAS	SET POINTER TO START OF
0396	07136	077165R		STB MAXC	CORE LOCATION
0397*					
0398	07137	162724R	BLIN1	LDA ALINK,I	GET THE LINK WORD
0399	07140	067165R		LDB MAXC	OUTPUT LINK TO
0400	07141	017445R		JSB LABD0	DISC BUFFER
0401	07142	037165R		ISZ MAXC	INCREMENT CORE ADDR POINTER
0402	07143	036724R		ISZ ALINK	INCREMENT LINK BUFFER POINTER
0403	07144	037152R		ISZ LWH2	LWH2 * = # WORDS IN BUFFER
0404	07145	027137R		JMP BLIN1	MORE TO OUTPUT
0405	07146	127132R		JMP BLIN0,I	DONE

0407	07147	000000	TEMP1	NOP	LOW BOUND OF LOW LINK AREA
0408	07150	000000	TEMP2	NOP	TEMPORARY
0409	07151	000000	LWH1	NOP	UPPER LINK AREA PRESENT FLAG
0410	07152	000000	LWH2	NOP	LENGTH OF CURRENT PAGE LINK AREA
0411	07153	000000	LWH3	NOP	HOLDS ORIGINAL TBREL
0412	07154	000000	LWH4	NOP	HOLDS ORIGINAL TPREL
0413	07155	000000	CPL	NOP	CURRENT PAGE LINK FLAG #0,CP LNK
0414	07156	000000	L001	NOP	PASS 1/2 FLAG
0415	07157	000000	L01	NOP	PASS 3 FLAG
0416	07160	003704R	CLINK	DEF LINKC	
0417	07161	000000	CPBAS	NOP	LOW BOUND OF CURRENT PAGE BUFFER
0418	07162	000000	CPLOC	NOP	CURRENT POINTER IN CP BUFFER
0419	07163	000000	CPMAX	NOP	UPPER BND OF CURRENT PAGE BUFFER
0420	07164	000000	CPADD	NOP	CURRENT PAGE ADDRESS OF BUFFER
0421	07165	000000	MAXC	NOP	MAX RELOCATION ADDRESS

0423	07166	177670	N72	DEC	=72	
0424	07167	177756	N19	DEC	=19	
0425	07170	000013	.11	DEC	11	
0426	07171	000014	.12	DEC	12	
0427	07172	000016	.14	DEC	14	
0428	07173	000020	.16	DEC	16	
0429	07174	000022	.18	DEC	18	
0430	07175	000177	.127	DEC	127	
0431	07176	000600	M600	OCT	600	
0432	07177	000700	M700	OCT	700	
0433	07200	001100	M1100	OCT	1100	
0434	07201	001777	O1777	OCT	1777	
0435	07202	002000	M2000	OCT	2000	
0436	07203	076000	M0760	OCT	076000	
0437	07204	174000	M1740	OCT	174000	
0438	07205	100000	MSIGN	OCT	100000	
0439	07206	020052	BLAST	OCT	20052	
0440	07207	020040	BLNKS	OCT	20040	
0441	07210	000040	BLANK	OCT	40	
0442	07211	020000	UBLNK	OCT	20000	
0443*						
0444	07212	027505	SLSHE	ASC	1,/E	CONTROL PARAMETER TERMINATOR
0445	07213	007214R	DADDR	DEF	++1	
0446	07214	022104	SDBP1	ASC	2,\$DBP	\$DBP1
0447	07216	030400		OCT	030400	
0448	07217	042105	CHRDE	ASC	2,DEBU	DEBUG
0449	07221	043400		OCT	043400	
0450	07222	022104	\$DBP2	ASC	2,\$DBP	\$DBP2
0451	07224	031000		OCT	031000	
0452	07225	022115	\$MEMR	ASC	2,\$MEM	\$MEMR
0453	07227	051000		OCT	051000	
0454	07230	042102	CHROB	ASC	2,DBKP	DBKPT
0455	07232	052000		OCT	052000	
0456	07233	042530	AEXEC	ASC	2,EXEC	
0457	07235	020000		OCT	020000	
0458	07236	007237R	DOSH0	DEF	++1	
0459	07237	051105		ASC	9,RELOCATING LOADER	
0460	07250	007251R	NAENT	DEF	++1	
0461	07251	047101		ASC	8,NAME/ENTRY ADDR	
0462*						
0463	07261	000054	COMMA	OCT	54	COMMA
0464	07262	000050	LPARN	OCT	50	LEFT PARENTHESIS
0465	07263	000051	RPARN	OCT	51	RIGHT PARENTHESIS
0466	07264	000060	M60	OCT	60	
0467	07265	000200	M200	OCT	200	
0468	07266	000300	M300	OCT	300	
0469	07267	000400	M400	OCT	400	
0470	07270	177600	M7600	OCT	177600	
0471	07271	000071	D57	DEC	57	
0472	07272	177600	N128	DEC	=128	
0473	07273	000200	.128	DEC	128	
0474*						
0475	07274	007326R	ADBUF	DEF	DBUF	ADDRESS OF DBUF BUFFER
0476	07275	007526R	AIBUF	DEF	IBUF	
0477	07276	007726R	ALBUF	DEF	LBUF	
0478	07277	000205R	AMEM5	DEF	MLIST+5	

0479	07300	000200R	AMLST	DEF	MLIST	
0480	07301	000157	DFUTS	DEF	UDNTS	
0481	07302	000160	DFSTS	DEF	SYNTS	
0482	07303	000214R	DFPBF	DEF	PBUF	ADDRESS OF NAME BUFFER
0483	07304	000000R	PLAD	DEF	PLIST	
0484*						
0485	07305	007306R	OPWRS	DEF	**+1	
0486	07306	010000		OCT	10000	
0487	07307	001000		OCT	1000	
0488	07310	000100		OCT	100	
0489	07311	000010		OCT	10	
0490	07312	000001		OCT	1	
0491*						
0492	07313	007314R	DPWRS	DEF	**+1	
0493	07314	023420		DEC	10000	
0494	07315	001750		DEC	1000	
0495	07316	000144		DEC	100	
0496	07317	000012		DEC	10	
0497	07320	000001		DEC	1	
0498*						
0499	07321	007322R	RBTAD	DEF	**+1	RELOCATION BASE TABLE
0500	07322	000000		OCT	0	ABSOLUTE PROGRAM BASE
0501	07323	000000	TPREL	OCT	0	CURRENT PROGRAM BASE
0502	07324	000000		OCT	0	BP RELOCATION ADDRESS
0503	07325	000000	COMAD	OCT	0	CURRENT COMMON RELOCATION BASE
0504*						
0505	07326	000000	DBUF	BSS	128	RELOCATABLE DISC INPUT BUFFER
0506	07526	000000	IBUF	BSS	128	ID SEGMENT BUFFER
0507	07726	000000	LBUF	BSS	512	
0508	00000		PLIST	EQU	LOADR	PUT 128 WD BUFFER AT START OF LO
0509	10726	000000	DATAB	BSS	36	PARAMETER INPUT BUFFER
0510	00200		MLIST	EQU	LOADR+128	MEMORY MAP BUFFER
0511	00210		TBUF	EQU	LOADR+136	TEMP BUFFER
0512	00214		PBUF	EQU	LOADR+140	
0513*						
0514	10772	000000	BLST	NOP		LOADER SYMBOL TABLE ADDRESS
0515	10773	000000	TLST	NOP		CURRENT LST ADDRESS
0516	10774	000000	PLST	NOP		END LST ADDRESS
0517*						
0518	10775	000000	LST1	NOP		LST NAME 1,2
0519	10776	000000	LST2	NOP		LST NAME 3,4
0520	10777	000000	LST3	NOP		LST NAME 5 ; ORDINAL
0521	11000	000000	LST4	NOP		PROGRAM IDENT ADDRESS
0522	11001	000000	LST5	NOP		BASE PAGE LINK ADDRESS
0523*						
0524	11002	000000	BIDNT	NOP		PROGRAM NAME TABLE ADDRESS
0525	11003	000000	TIDNT	NOP		CURRENT IDENT ADDRESS
0526	11004	000000	PIDNT	NOP		END IDENT ADDRESS
0527	11005	000000	CIDNT	NOP		WORKING IDENT ADDRESS
0528*						
0529	11006	000000	ID1	NOP		IDENT NAME 1,2
0530	11007	000000	ID2	NOP		IDENT NAME 3,4
0531	11010	000000	ID3	NOP		IDENT NAME 5 ; USEAGE FLAG
0532	11011	000000	ID4	NOP		IDENT COMMON LENGTH
0533	11012	000000	ID5	NOP		IDENT DISC ADDRESS
0534	11013	000000	ID6	NOP		(15)=MAIN/SUBROUTINE; (6=0)=TYPE

0535	11014	000000	ID7	NOP	DISC LENGTH; MAIN IDENT FOR SEGM
0536*					
0537	11015	000000	ABSID	NOP	IDENT ADDR FOR NEXT SEGMENT SCAN
0538	11016	000000	AMAD	NOP	CURRENT MLIST ADDRESS
0539*					
0540	11017	000000	BPEND	NOP	END OF BASE PAGE BUFFER
0541	11020	000000	BP#	NOP	# OF BASE PAGE LINK WORDS
0542	11021	000000	BSBAD	NOP	SEGMENT BP RELOC ADDR
0543	11022	000000	BSPAD	NOP	SEGMENT PROG RELOC ADDR
0544	11023	000000	BSSDP	NOP	INITIAL DISC RES BSS DISPLACEMENT
0545	11024	000000	BTRSC	NOP	CURRENT UTIL PROG T/S ADDRESS
0546	11025	000000	BGBND	NOP	USER PROG BOUNDARY
0547	11026	000000	BPOIT	NOP	NEG OF BP STARTING ADDRESS
0548*					
0549	11027	000000	CBASE	NOP	CURRENT BASE PROGRAM ADDRESS
0550	11030	000000	CLXXX	NOP	
0551	11031	000000	CNFLG	NOP	TEMPORARY FLAG
0552	11032	000000	COUNT	NOP	# OF BINARY SECTOR COUNTER
0553	11033	000000	CURAD	NOP	CURRENT DBUF ADDRESS
0554	11034	000000	CURAI	NOP	CURRENT IBUF ADDRESS
0555	11035	000000	CURAL	NOP	CURRENT LBUF ADDRESS
0556	11036	000000	CURAP	NOP	CURRENT PLIST ADDRESS
0557	11037	000000	CURAT	NOP	CURRENT TBUF ADDRESS
0558*					
0559	11040	000000	DBASE	NOP	CURRENT BASE DISC ADDRESS
0560	11041	000000	DBFLG	NOP	NO DEBUG/DEBUG OPTION
0561	11042	000000	DBLAD	NOP	CURRENT DBL ADDRESS
0562	11043	000000	DEBUGF	NOP	SPECIAL DEBUG ENTRY DEBUG FLAG
0563	11044	000000	DCNT	NOP	CURRENT DBUF COUNT
0564	11045	000000	DCRNT	NOP	CURRENT STORE DISC ADDRESS
0565	11046	000000	DISCX	NOP	WHICH DISC? FLAG
0566	11047	000000	DSKAB	NOP	STARTING ABS PROG DISC ADDRESS
0567	11050	000000	DSKAD	NOP	CURRENT DISC ADDRESS
0568	11051	000000	DSKBR	NOP	CURRENT MAIN DISC RES DISC ADDR
0569	11052	000000	DSKBS	NOP	DISC ADDR OF MAIN DISC RES BP
0570	11053	000000	DSKID	NOP	ID SEGMENT DISC ADDRESS
0571	11054	000000	DSKMN	NOP	INITIAL MAIN DISC ADDRESS
0572	11055	000000	DSKRD	NOP	DISC INPUT ADDRESS
0573	11056	000000	DSKSS	NOP	BINARY SCRATCH ADDRESS
0574	11057	000000	DSKTM	NOP	TEMPORARY DISC ADDRESS
0575	11060	000000	DSK01	NOP	
0576	11061	000000	DSK02	NOP	
0577	11062	000000	DSKUD	NOP	
0578	11063	000000	DSCNT	NOP	# SECTORS FOR CURRENT PROGRAM
0579*					
0580	11064	000000	ETFLG	NOP	END OF TAPE FLAG
0581	11065	000000	EXCNT	NOP	TEMPORARY COUNTER
0582	11066	000000	EXORD	NOP	EXTERNAL ORDINAL
0583*					
0584	11067	000000	FIOPT	NOP	FILE SEARCH OPTION
0585	11070	000000	HDFLG	NOP	HEADING FORMAT FLAG
0586*					
0587	11071	000000	ICNT	NOP	CURRENT IBUF COUNT
0588	11072	000000	IDMBS	NOP	USER MAIN ADDR FOR SEGMENT REF
0589	11073	177777	IGNOR	DEC	LOADING/NOT LOADING FLAG
0590	11074	000000	IMAIN	NOP	MAIN PROG IDENT ADDRESS

0591	11075	000000	INSCN NOP	INSTRUCTION TYPE COUNTER
0592	11076	000000	INSTR NOP	CURRENT INSTRUCTION
0593	11077	000000	LCNT NOP	CURRENT LBUF COUNT
0594	11100	000000	LDING NOP	PROGRAMS LOADING FLAG
0595	11101	000000	LOWER NOP	LOWER BP LST BOUND
0596	11102	000000	MAXAD NOP	CURRENT MAX PROG LOAD ADDRESS
0597*				
0598	11103	000000	NBPBF NOP	NEG BASE PAGE BUFFER ADDRESS
0599	11104	000000	NXFLG NOP	ENT/EXT FLAG
0600*				
0601	11105	000000	OPRND NOP	CURRENT OPERAND
0602*				
0603	11106	000000	PAGNO NOP	CURRENT PAGE NO.
0604	11107	000000	PBREL NOP	INITIAL BP RELOCA ADDR
0605	11110	000000	PCNTT NOP	PROGRAM NAME COUNTER
0606	11111	000000	PGCNT NOP	# PROGRAMS LOADED COUNTER
0607	11112	000000	PGMIN NOP	INPUT UNIT (SECONDARY)
0608	11113	000000	PLCNT NOP	CURRENT PLOST COUNT
0609	11114	000000	PLFLG NOP	TEMPORARY FLAG
0610	11115	000000	PLGTH NOP	PROGRAM LENGTH
0611	11116	000000	PLMAN NOP	CURRENT MAIN PLIST ADDRESS
0612	11117	000000	PPREL NOP	INITIAL PROG RELOC ADDR
0613	11120	000000	PRENT NOP	PRIMARY ENTRY POINT
0614	11121	000000	PTYPE NOP	CURRENT PROGRAM TYPE
0615*				
0616	11122	000000	RANAD NOP	CURRENT POWER RANGE ADDRESS
0617	11123	000000	RELAD NOP	CURRENT CORE RELOCATION ADDRESS
0618	11124	000000	REKEY NOP	INSTRUCTION TYKPE BYTE
0619	11125	000000	RIC NOP	RECORD ID CODE
0620	11126	000000	RLGTH NOP	RELOCATABLE UTIL REC LENGTH
0621*				
0622	11127	000000	SAVSS NOP	SAVE SS FLAG
0623	11130	000000	SYM12 NOP	SYMBOL ADDRESS 1
0624	11131	000000	SYM34 NOP	SYMBOL ADDRESS 2
0625	11132	000000	SYM5 NOP	SYMBOL ADDRESS 3
0626*				
0627	11133	000000	TABS NOP	ABSOLUTE WORD
0628	11134	000000	TADR NOP	ABSOLUTER ADDRESS
0629	11135	000000	TBREL NOP	CURRENT BP RELOC ADDR
0630	11136	000000	TCNT NOP	CURRENT TBUF COUNT
0631	11137	000000	TDIR NOP	BINARY READ DIRECTION FLAG
0632	11140	000000	TEMP NOP	TEMPORARY
0633	11141	000000	TERMN NOP	TERMINATOR FLAG
0634	11142	000000	TNEXT NOP	NEXT TRACK SECTOR ADDRESS
0635	11143	000000	TTYNO NOP	TTY #
0636	11144	000000	TWORD NOP	WORD IN BUFFER ADDRESS
0637*				
0638	11145	000000	UPPER NOP	UPPER BP LST BOUND
0639	11146	000000	UTCNT NOP	# OF UTIL RECORDS LEFT (-)
0640	11147	000000	UTRSC NOP	CURRENT UTIL REC T/S ADDRESS
0641*				
0642	11150	000000	WDCNT NOP	TEMPORARY COUNTER
0643*				
0644	11151	000000	XPREL NOP	TEMPORARY COUNTER
0645*				
0646	11152	000000	UBFWA NOP	LOWER BASE PAGE BOUND

PAGE 0101 #07 DISC OPERATING SYSTEM RELOCATING LOADER

0647	11153	000000	UBLWA NOP	UPPER BASE PAGE BOUND
0648	11154	000000	UMFWA NOP	LOWER MAIN PROG BOUND
0649	11155	000000	UMLWA NOP	UPPER MAIN PROG BOUND
0650	11156	000000	DVR NOP	DEVICE DRIVER FLAG

0652	11157	000606	LISTU	OCT	606
0653	11160	000000	ZBUF	BSS	3
0654	11163	030061	ERR01	ASC	1,01
0655	11164	030062	ERR02	ASC	1,02
0656	11165	030063	ERR03	ASC	1,03
0657	11166	030064	ERR04	ASC	1,04
0658	11167	030065	ERR05	ASC	1,05
0659	11170	030066	ERR06	ASC	1,06
0660	11171	030067	ERR07	ASC	1,07
0661	11172	030070	ERR08	ASC	1,08
0662	11173	030071	ERR09	ASC	1,09
0663	11174	030460	ERR10	ASC	1,10
0664	11175	030461	ERR11	ASC	1,11
0665	11176	030462	ERR12	ASC	1,12
0666	11177	030463	ERR13	ASC	1,13
0667	11200	030464	ERR14	ASC	1,14
0668	11201	030465	ERR15	ASC	1,15
0669	11202	030466	ERR16	ASC	1,16
0670	11203	047117	ASCNO	ASC	1,NO
0671	11204	000331	XSSFL	DEF	SSFLG
0672	11205	011206R	BPBUF	DEF	BPAGE
0673	11206		BPAGE	BSS	0

LIST OUTPUT UNIT
 CURRENT PROGRAM NAME BUFFER
 CHECKSUM ERROR
 ILLEGAL RECORD
 MEMORY OVERFLOW
 LINKAGE AREA OVERFLOW
 LOADER SYMBOL TABLE OVERFLOW
 DUPLICATE PROGRAM NAMES
 DUPLICATE ENTRY POINTS
 NO TRANSFER ADDRESS PRESENT
 RECORD OUT OF SEQUENCE

PROGRAM IDENT TABLE OVERFLOW
 SPECIFIED USER FILE NOT IN DIRECT
 DUPL PROGRAM NAME
 NON-ZERO BASE PAGE LENGTH
 SEGMENT LOADED BEFORE MAIN,
 PROGRAM OVERLAY

ADDRESS OF SS FLAG IN BASE PAGE
 ADDR OF BASE PAGE BUFFER
 BEGINNING OF BASE PAGE BUFFER

0675	00000	A	EQU 0	A REGISTER
0676	00001	B	EQU 1	B REGISTER
0677	00100	.	EQU 100B	
0678	00053	..	EQU 53B	
0679	00041	N10	EQU ..-10	DEC -10
0680	00042	N9	EQU ..-9	DEC -9
0681	00043	N8	EQU ..-8	DEC -8
0682	00044	N7	EQU ..-7	DEC -7
0683	00045	N6	EQU ..-6	DEC -6
0684	00046	N5	EQU ..-5	DEC -5
0685	00047	N4	EQU ..-4	DEC -4
0686	00050	N3	EQU ..-3	DEC -3
0687	00051	N2	EQU ..-2	DEC -2
0688	00052	N1	EQU ..-1	DEC -1
0689	00053	N0	EQU ..	DEC 0
0690	00054	.1	EQU ..+1	DEC 1
0691	00055	.2	EQU ..+2	DEC 2
0692	00056	.3	EQU ..+3	DEC 3
0693	00057	.4	EQU ..+4	DEC 4
0694	00060	.5	EQU ..+5	DEC 5
0695	00061	.6	EQU ..+6	DEC 6
0696	00062	.7	EQU ..+7	DEC 7
0697	00063	.8	EQU ..+8	DEC 8
0698	00064	.9	EQU ..+9	DEC 9
0699	00065	.10	EQU ..+10	DEC 10
0700	00066	.17	EQU ..+11	DEC 17
0701	00070	M17	EQU ..+13	OCT 17
0702	00071	M37	EQU ..+14	OCT 37
0703	00072	M77	EQU ..+15	OCT 77
0704	00073	M177	EQU ..+16	OCT 177
0705	00074	M377	EQU ..+17	OCT 377
0706	00075	M1774	EQU ..+18	OCT 177400
0707	00076	M3777	EQU ..+19	OCT 3777
0708	00077	M1777	EQU ..+20	OCT 177700
0709	00100	SMLWA	EQU ..+0	LWA OF USEABLE MEMORY
0710	00101	JBINS	EQU ..+1	START T/S OF JBIN AREA
0711	00102	JBINC	EQU ..+2	CURRENT T/S OF JBIN AREA
0712	00111	BATCH	EQU ..+9	L.U. OF BATCH INPUT DEVICE
0713	00112	SYSTY	EQU ..+10	L.U. OF SYSTEM TELETYPE
0714	00116	SECTR	EQU ..+14	# OF SECTORS/TRACK
0715	00147	EXP67	EQU ..+39	
0716	00154	DISCO	EQU ..+44	LAST TRACK ON DISC
0717	00155	SYSSC	EQU ..+45	
0718	00157	UDNTS	EQU ..+47	NEXT USER AVAILABLE T/S
0719	00160	SYNTS	EQU ..+48	NEXT SYSTEM AVAILABLE T/S
0720	00161	CUDSC	EQU ..+49	CURRENT USER DISC SUBCHANNEL
0721	00174	TSONE	EQU ..+60	NEXT T/S AFTER LAST REQUEST
0722	00200	DISCL	EQU ..+64	=10 IF USER AREA NOT ON SYS DISC
0723	00254	SMFWA	EQU ..+108	FWA OF USER MAIN AREA
0724	00255	SBFWA	EQU ..+109	FWA OF USER BASE AREA
0725	00256	SBLWA	EQU ..+110	LWA OF USER BASE AREA
0726	00313	MSECT	EQU ..+139	NEG # OF SECTORS/TRACK
0727	00324	DSCLB	EQU ..+148	RELOCATABLE LIBRARY T/S
0728	00325	DSCL#	EQU ..+149	# OF RELOCATABLE LIBRARY SECTORS
0729	00331	SSFLG	EQU ..+153	
0730			END LOADR	

PAGE 0104 #07 DISC OPERATING SYSTEM RELOCATING LOADER

** NO ERRORS*

#LIB	00387/02	00252/02	00383/02				
#SECT	00505/01	00483/01	00487/01				
#WRDS	00302/05	00141/03	00144/03	00171/03	00200/03	00276/03	00281/03
	00296/05						
#SDBP1	00446/07						
#SDBP2	00450/07						
#SMEMR	00452/07						
.	00677/07	00709/07	00710/07	00711/07	00712/07	00713/07	00714/07
	00715/07	00716/07	00717/07	00718/07	00719/07	00720/07	00721/07
	00722/07	00723/07	00724/07	00725/07	00726/07	00727/07	00728/07
	00729/07						
..	00678/07	00679/07	00680/07	00681/07	00682/07	00683/07	00684/07
	00685/07	00686/07	00687/07	00688/07	00689/07	00690/07	00691/07
	00692/07	00693/07	00694/07	00695/07	00696/07	00697/07	00698/07
	00699/07	00700/07	00701/07	00702/07	00703/07	00704/07	00705/07
	00706/07	00707/07	00708/07				
.1	00690/07	00154/01	00291/01	00339/01	00344/01	00421/01	00562/01
	00023/02	00235/02	00263/02	00348/03	00212/04	00266/04	00142/05
	00020/07	00026/07					
.10	00699/07	00548/01	00534/03	00688/05			
.11	00425/07	00522/03					
.12	00426/07	00341/01					
.127	00430/07	00648/05	00659/05				
.128	00473/07	00480/01	00565/01	00143/03	00199/03	00256/03	00275/03
	00280/03	00376/03	00234/05	00334/05	00516/05	00341/06	00346/06
.14	00427/07	00015/04	00408/05				
.16	00428/07	00290/01	00571/03	00202/04	00439/04	00544/06	
.17	00700/07	00216/01					
.18	00429/07	00364/01	00466/03	00567/03	00011/04		
.2	00691/07	00308/01	00338/01	00433/01	00466/01	00511/01	00521/01
	00009/02	00027/02	00211/02	00233/02	00279/02	00513/03	00151/04
	00251/04	00141/05	00180/05	00218/05	00556/06	00019/07	00083/07
.3	00692/07	00468/01	00029/02	00281/02	00561/02	00564/02	00006/03
	00068/03	00155/03	00159/03	00197/03	00329/03	00450/03	00460/03
	00525/03	00564/03	00581/03	00001/04	00148/04	00253/04	00325/04
	00071/06	00562/06					

.4	00693/07	00470/01	00470/01	00031/02	00283/02	00452/03	00255/04
	00281/04	00528/04	00152/05	00630/05	00631/05	00675/05	
.5	00694/07	00375/01	00458/01	00472/01	00033/02	00120/02	00285/02
	00530/02	00015/03	00043/03	00050/03	00454/03	00474/03	00528/03
	00623/03	00637/03	00168/04	00257/04	00512/04	00530/04	00327/05
	00411/05	00642/05	00654/05				
.6	00695/07	00059/04	00216/04	00229/04	00470/04		
.7	00696/07	00553/01	00222/02	00341/03	00211/04	00231/04	00250/04
	00472/04	00527/04	00156/07				
.8	00697/07	00022/07					
.9	00698/07	00519/03	00226/04				
A	00675/07	00209/04	00217/04	00227/04	00248/04	00304/04	00533/04
	00127/05	00383/05	00432/05	00445/05	00448/05	00754/05	00067/07
	00161/07	00254/07	00260/07	00271/07	00285/07		
AABUF	00522/05	00534/01	00192/05	00202/05	00502/05		
ABBUF	00523/05	00195/05	00200/05	00505/05			
ABFLG	00495/05	00538/01	00193/05	00199/05	00481/05	00503/05	00509/05
ABORT	00544/06	00378/01	00533/02	00018/03	00484/03	00124/05	00618/06
	00625/06						
ABRTX	00543/06	00037/02	00040/02	00288/02	00578/02	00011/03	00511/03
	00284/04	00454/05	00020/06	00056/06			
ABSID	00537/07	00647/03	00668/03				
ABUF	00524/05	00522/05					
ADBUF	00475/07	00146/03	00158/03	00167/03	00577/03	00044/04	00049/04
	00041/06	00107/06	00111/06	00131/06	00147/06	00151/06	00164/06
AEXEC	00456/07	00259/01	00261/01	00263/01	00162/02	00166/02	00172/02
AIBUF	00476/07	00491/03	00497/03	00548/03	00028/04	00771/05	00773/05
ALBUF	00477/07	00580/01	00095/02	00250/03	00253/03		
ALINK	00233/07	00219/04	00236/07	00296/07	00394/07	00398/07	00402/07
ALLOC	00440/05	00448/04	00389/05	00452/05			
AMAD	00538/07	00431/01	00434/01	00438/01	00440/01	00444/01	00448/01
	00451/01	00452/01	00453/01	00124/04	00128/04	00130/04	00131/04
	00134/04	00135/04	00141/04	00399/06	00403/06		
AMEM5	00478/07	00196/04	00437/04				

AMLST 00479/07 00123/04 00203/04 00440/04 00397/06
ANEW 00167/05 00164/05
ASCNO 00670/07 00169/01 00052/04
B 00676/07 00136/01 00141/01 00147/01 00150/01 00153/01 00206/01
00223/01 00435/01 00441/01 00446/01 00554/01 00010/02 00017/02
00223/02 00227/02 00524/02 00620/02 00625/02 00160/03 00342/03
00358/03 00514/03 00182/04 00188/04 00239/04 00241/04 00244/04
00301/04 00316/04 00350/04 00353/04 00359/04 00372/04 00377/04
00405/04 00408/04 00479/04 00494/04 00498/04 00566/04 00584/04
00591/04 00593/04 00598/04 00029/05 00251/05 00255/05 00257/05
00313/05 00576/05 00579/05 00600/05 00609/05 00633/05 00636/05
00046/06 00060/06 00062/06 00094/06 00116/06 00125/06 00215/06
00222/06 00268/06 00273/06 00307/06 00314/06 00327/06 00363/06
00401/06 00451/06 00471/06 00478/06 00166/07 00249/07 00330/07
B1777 00632/06 00599/06
BATCH 00712/07 00328/01 00332/01 00615/06 00622/06 00008/07 00012/07
BBADD 00021/07 00005/07
BBUF 00525/05 00523/05
BD005 00015/04 00010/04
BD010 00018/04 00014/04
BDEND 00007/04 00594/03
BDLD 00593/03 00597/03 00006/04
BGBND 00546/07 00554/03
BGSPL 00174/04 00169/04
BIDNT 00524/07 00199/01 00585/03 00082/05
BLANK 00441/07 00140/04 00434/04 00061/07
BLAST 00439/07 00426/04
BLCK# 00286/03 00198/03 00242/03 00255/03 00260/03 00274/03 00356/03
BLCK1 00271/03 00246/03
BLCK2 00248/03 00278/03
*BLCK3 00266/03
BLCK5 00263/03 00268/03
BLCK6 00280/03 00268/02 00385/02 00273/03

BLIND 00392/07 00349/07 00386/07 00405/07
BLIN1 00398/07 00404/07
*BLIN2 00350/07
BLIN3 00353/07 00102/04 00341/07 00366/07
BLINK 00376/07 00312/04 00313/07 00319/07 00322/07 00326/07 00336/07
00378/07 00385/07 00390/07
BLKAD 00285/03 00414/01 00257/02 00362/02 00239/03 00249/03 00264/03
00266/03
BLNKS 00440/07 00192/04 00400/06 00439/06
BLOCK 00237/03 00416/01 00260/02 00269/03 00411/03
BLST 00514/07 00191/01 00044/05
BMES 00633/06 00592/06
BMES1 00635/06 00603/06
BP# 00541/07 00187/01 00201/01
BPADD 00435/05 00428/05 00433/05
BPAGE 00673/07 00672/07
BPALC 00388/05 00377/05
BPBE 00613/06 00595/06 00601/06
BPBUF 00672/07 00179/01 00188/01 00193/01 00204/01 00027/06 00381/05
00430/05 00443/05 00753/05 00311/06 00329/06
BPEND 00540/07 00189/01
BPN 00236/04 00224/04 00233/04
BPOIT 00547/07 00185/01 00026/06 00380/05 00429/05 00442/05 00752/05
00312/06 00330/06
BPOUT 00325/06 00618/03 00660/03 00677/03 00333/06 00344/06
BPSCN 00368/05 00419/04 00589/04 00387/05 00390/05 00299/07
BPSRC 00374/05 00385/05
BPSTR 00077/06 00174/01 00035/04 00042/04 00056/04 00081/06
BPSYO 00334/06 00348/06
BSBAD 00542/07 00622/03 00650/03 00661/03 00664/03 00675/03 00681/03

00655/05

BSEND 00671/03 00630/03

BSLD 00629/03 00635/03 00639/03 00645/03 00670/03

BSPAD 00543/07 00620/03 00666/03 00643/05

BSSDP 00544/07 00515/04 00644/05

*BTRSC 00545/07

BUFCL 00309/05 00536/01 00227/05 00317/05 00775/05 00108/06 00148/06

BUMP 00401/03 00310/02 00311/02 00314/02 00317/02 00320/02 00347/02
00479/02 00480/02 00483/02 00486/02 00490/02 00493/02 00496/02
00499/02 00615/02 00616/02 00673/02 00674/02 00678/02 00681/02
00685/02 00086/03 00337/03 00409/03 00412/03 00432/03

CBASE 00549/07 00517/04 00021/06

CBLA 00287/03 00240/03 00363/03 00371/03

CBUF 00515/05 00506/05 00539/05

CDEC 00758/05 00729/05

CF10 00238/02 00234/02

CFILE 00233/02 00225/02

CHKER 00042/02 00019/02

*CHKOK 00339/03

CHRDB 00464/07 00201/06 00205/06 00210/06

CHRDE 00448/07 00145/02 00147/02 00149/02 00341/04 00345/04 00351/04
00610/04 00614/04 00002/05 00009/05 00013/05 00018/05 00189/06
00193/06 00198/06

CIDNT 00527/07 00586/03 00416/06 00423/06

CL 00226/05 00221/05

CLBPS 00304/06 00292/06

CLFLG 00324/04 00331/04 00334/04

CLID3 00326/05 00003/04 00327/04 00339/05 00343/05

CLINK 00416/07 00300/07

CLIST 00396/06 00117/04 00425/04 00272/06 00275/06 00405/06

CLORD 00376/02 00381/02

CLR0 00458/03 00481/03
CLRID 00443/03 00457/03 00471/03 00478/03
CLRLT 00285/06 00663/03 00682/03 00110/04 00309/06 00318/06
CLRNX 00291/06 00296/06 00300/06 00303/06
CLRTK 00085/06 00040/06 00133/06
CLSRC 00246/04 00497/04 00546/04
CLST 00289/04 00294/04
CLSUT 00354/05 00359/05 00361/05
CLXXX 00550/07 00332/04 00335/04 00181/06 00182/06
CMES 00389/01 00340/01
CMES8 00063/04 00016/04
CMND 00494/05 00477/05 00514/05
CNFLG 00551/07 00064/02 00084/02 00111/02
COMAD 00503/07 00555/03 00574/04
COMMA 00463/07 00458/06 00063/07
CON 00153/07 00146/07 00149/07 00163/07
CONV 00137/07 00104/07 00119/07 00177/07
CONV1 00143/07 00151/07
CONV2 00165/07 00155/07
CONV3 00194/07 00166/07
CONVD 00355/06 00197/04 00438/04 00379/06
COREA 00295/05 00269/05 00282/05
COUNT 00552/07 00282/01 00302/01 00304/01 00366/01 00370/01 00255/02
00244/03 00248/03 00271/03
CPADD 00420/07 00295/07 00307/07 00362/07
CPBAS 00417/07 00293/07 00314/07 00360/07 00395/07
CPL 00413/07 00151/01 00244/07
CPLOC 00418/07 00292/07 00310/07 00329/07 00334/07 00359/07 00371/07
00380/07 00388/07

CPLST 00181/04 00173/04
CPMAX 00419/07 00287/07 00331/07 00364/07
CPXXX 00187/04 00180/04
CROSS 00339/07 00264/04 00310/07
CUDSC 00720/07 00240/01 00037/04
CURAD 00553/07 00535/01 00102/02 00107/02 00578/03 00203/05 00226/05
CURAI 00554/07 00492/03 00499/03 00512/03 00520/03 00526/03 00765/05
00766/05 00774/05
CURAL 00555/07 00432/01 00581/01 00007/02 00008/02 00015/02 00096/02
00101/02 00108/02 00300/02 00312/02 00315/02 00318/02 00481/02
00484/02 00487/02 00491/02 00494/02 00497/02 00500/02 00611/02
00617/02 00668/02 00676/02 00679/02 00682/02 00126/03 00254/03
00317/03 00339/03 00403/03 00207/04 00215/04 00225/04 00236/04
00240/04 00242/04 00247/04 00265/04 00271/04 00383/04 00386/04
00389/04 00415/04 00504/04 00519/04 00534/04 00551/04 00563/04
00568/04 00575/04 00035/05 00461/05 00540/05
CURAP 00556/07 00588/03 00625/03 00005/04 00363/04 00366/04 00331/05
00337/05 00340/05 00224/06 00226/06
CURAT 00557/07 00331/06 00336/06 00345/06 00347/06 00356/06 00368/06
00369/06 00372/06 00374/06 00375/06 00376/06
D57 00471/07 00333/03
DADDR 00445/07 00229/06
DATAB 00509/07 00346/01 00355/01 00028/07
DATE 00346/01 00352/01 00037/07
DBASE 00559/07 00028/06 00170/06 00260/06
DBFLG 00560/07 00148/01 00138/02 00329/04
DBL 00609/02 00497/01 00074/02 00294/02 00622/02 00628/02
DBL0 00518/04 00507/04
DBL1 00519/04 00550/04
DBL2 00524/04 00549/04
DBL3 00535/04 00556/04 00567/04 00602/04
DBL4 00551/04 00529/04
DBL5 00555/04 00561/04

DBL6 00568/04 00531/04
DBL7 00598/04 00587/04
DBL8 00601/04 00597/04
DBLAD 00561/07 00505/04 00535/04 00547/04 00581/04
DBLR 00074/02 00030/02
DBSET 00460/05 00246/04 00268/04 00269/04 00381/04 00382/04 00385/04
00388/04 00391/04 00442/04 00502/04 00503/04 00518/04 00523/04
00543/04 00570/04 00463/05 00465/05
DBUF 00505/07 00150/03 00151/03 00157/03 00164/03 00047/04 00475/07
DEBUG 00562/07 00021/06 00049/06 00058/06
DCNT 00563/07 00533/01 00103/02 00114/02 00576/03 00213/05
DCRNT 00564/07 00035/06 00039/06 00091/06 00095/06 00117/06 00130/06
00163/06 00171/06
DDBLR 00498/04 00254/04
DDOUT 00190/05 00105/02 00117/02 00238/05
DDRV 00480/03 00453/03
DEBUG 00608/04 00287/04 00612/04 00616/04 00004/05 00038/05
DEI10 00338/04 00343/04 00347/04 00355/04
DEIDG 00329/04 00318/04
DEN10 00286/04 00278/04 00282/04
DEN20 00287/04 00275/04
DENDR 00260/04 00258/04
DENTR 00374/04 00252/04
DEXTR 00375/04 00256/04
DFPBF 00482/07 00376/01 00446/06
DFSTS 00481/07 00041/04
DFUTS 00480/07 00034/04
DINIT 00168/06 00605/03 00650/03 00175/06
DISCL 00722/07 00488/03 00043/04 00048/04

DISCO 00716/07 00242/01

DISCX 00565/07 00309/01 00512/01 00212/02 00545/02 00152/04 00294/05
00473/05 00072/06

DISKA 00249/05 00209/05 00256/05 00260/05 00594/05 00031/06 00261/06

DISKI 00268/05 00147/03 00251/03 00498/03 00045/04 00273/05 00132/06

DISKO 00281/05 00168/03 00549/03 00029/04 00050/04 00286/05 00772/05
00112/06 00152/06 00165/06 00337/06

DLIN1 00326/07 00318/07 00321/07

DLIN2 00319/07 00324/07

DLINK 00298/07 00388/05 00302/07 00305/07 00328/07 00332/07 00354/07

DLU 00492/05 00219/05 00474/05 00558/05

DONE 00116/07 00090/07 00095/07 00101/07 00118/07

DONE1 00119/07 00124/07

DOSH0 00458/07 00568/03

DPWRS 00492/07 00359/06

DRCH0 00505/03 00531/03

DRCH1 00512/03 00504/03

DRCH5 00525/03 00518/03

DRCH6 00530/03 00524/03

DRCHA 00510/03 00226/01 00129/02 00780/05 00090/06

DRCHK 00499/03 00532/03

DRCLR 00547/03 00541/03

DRKEY 00174/05 00569/03 00573/03 00204/04 00441/04 00165/05 00184/05

DRSCH 00491/03 00509/03

DRWOW 00500/05 00486/05 00520/05 00546/05

DSCAN 00006/05 00011/05 00015/05 00020/05

DSCL# 00728/07 00250/02

DSCLB 00727/07 00256/02

DSCNT 00578/07 00063/02 00122/02 00210/02 00263/03 00224/05 00480/05
00544/05

DSK01 00575/07 00097/06 00124/06 00142/06
DSK02 00576/07 00102/06 00128/06 00146/06
DSKAB 00566/07 00248/01 00579/03
DSKAD 00567/07 00217/01 00220/01 00231/01 00235/01 00277/01 00367/01
00412/01 00515/01 00545/01 00049/02 00066/02 00128/02 00580/03
00614/03 00671/03 00674/03 00679/03 00033/04 00040/04 00163/04
00205/05 00208/05 00210/05 00169/06 00264/06 00336/06 00339/06
DSKBR 00568/07 00672/03 00678/03
DSKBS 00569/07 00615/03 00673/03
DSKID 00570/07 00490/03 00496/03 00506/03 00547/03 00025/04 00770/05
00777/05
DSKNN 00571/07 00166/04 00638/05
DSKNM 00587/02 00413/01 00067/02 00258/02 00361/02 00551/02 00367/03
DSKRD 00572/07 00471/05 00483/05 00532/05 00534/05
DSKSS 00573/07 00237/01 00514/01
DSKTM 00574/07 00246/01 00088/06
DSKUD 00577/07 00110/06 00113/06 00115/06 00154/06
DTRTM 00132/02 00230/01 00127/02
DUP 00013/03 00003/03
DVR 00660/07 00480/03 00280/04 00629/05 00674/05
DVR1 00724/05 00731/05 00734/05 00738/05 00745/05 00750/05
DVRAD 00712/05 00676/05 00725/05
DVRS 00741/05 00732/05 00736/05 00755/05
DWRIT 00094/02 00071/02 00085/02 00130/02
EBUF 00233/05 00196/05
EN.XT 00665/02 00502/01 00079/02 00089/03
END 00123/03 00475/01 00087/02 00132/03
END1 00383/02 00377/02
ENDDR 00533/03 00501/03
ENDR 00082/02 00034/02

ENEX1 00393/04 00398/04 00402/04 00410/04
ENEX2 00335/02 00322/02
ENEX3 00343/02 00328/02
ENEX4 00351/02 00337/02
ENTID 00360/04 00320/04
ENTR 00077/02 00028/02
ENTXX 00399/02 00404/02 00408/02 00416/02
ENXT1 00084/03 00022/03 00052/03 00060/03 00080/03
ENXX1 00332/02 00357/02
EPLST 00321/04 00369/04
ERR01 00654/07 00042/02
ERR02 00655/07 00036/02
ERR03 00656/07 00019/06
ERR04 00657/07 00453/05
ERR05 00658/07 00542/06
ERR06 00659/07 00472/03
ERR07 00660/07 00013/03
ERR08 00661/07 00283/04
ERR09 00662/07 00039/02
ERR10 00663/07 00510/03
ERR11 00664/07 00122/05
ERR12 00665/07 00373/01
ERR13 00666/07 00528/02
ERR14 00667/07 00577/02
ERR15 00668/07 00010/03
ERR16 00669/07 00055/06
ERR17 00290/02 00287/02
ERR18 00630/06 00613/06

ERR19 00631/06 00620/06

ERROR 00150/05 00374/01 00043/02 00529/02 00014/03 00473/03 00123/05
00155/05 00543/06 00614/06 00621/06

ETFLG 00580/07 00558/01 00572/01 00578/01

EXCNT 00581/07 00158/02 00194/02 00198/02 00209/02 00214/02 00372/02
00671/02 00087/03 00380/04 00495/04 00501/04 00544/04 00398/05
00405/05 00414/05 00416/05 00470/06 00477/06

EXEC 00004/01 00214/01 00288/01 00336/01 00342/01 00362/01 00551/01
00560/01 00220/02 00464/03 00562/03 00057/04 00139/05 00178/05
00229/05 00291/05 00511/05 00555/05 00078/06 00560/06 00017/07
00024/07

EXEND 00495/04 00443/04 00453/04 00458/04 00466/04 00475/04 00484/04
00488/04 00491/04

EXNND 00345/02 00331/02 00334/02 00353/02 00360/02

EXORD 00582/07 00553/04 00559/04

EXP67 00715/07 00178/01

EXT1 00048/03 00708/02

EXT2 00444/04 00413/04

EXTR 00078/02 00032/02

FETCH 00315/03 00420/01 00457/01 00465/01 00004/02 00262/02 00278/02
00350/03 00369/03 00381/03

FIOPT 00584/07 00321/01 00238/02

FIRST 00241/05 00539/01 00215/05 00220/05

FLLA 00369/07 00345/07

GENID 00616/05 00608/03 00653/03 00686/05 00690/05 00696/05 00701/05

GET10 00449/06 00483/06

GET15 00486/06 00465/06

GET20 00487/06 00459/06 00461/06 00463/06

GETD 00380/06 00366/06 00370/06 00373/06 00392/06

GETL 00107/02 00104/02

GETP 00438/06 00358/01 00445/06 00485/06 00488/06 00489/06 00508/06
00529/06

GOCHK	00220/02	00177/02	00237/02				
GTS	00574/05	00206/05	00270/05	00283/05	00484/05	00535/05	00584/05
HABSS	00515/04						
HDFLG	00585/07	00649/03	00073/04	00125/04	00160/04	00273/04	
I.O	00288/05	00272/05	00285/05	00299/05			
IBUF	00506/07	00476/07					
ICNT	00587/07	00494/03	00521/03	00523/03	00527/03	00529/03	00533/03
	00539/03	00684/05	00687/05	00693/05	00767/05	00782/05	
ID1	00529/07	00332/02	00514/02	00531/02	00538/02	00020/03	00042/03
	00054/03	00462/03	00475/03	00603/03	00631/03	00129/04	00181/04
	00455/04	00097/05	00357/05	00623/05	00188/06	00200/06	00217/06
	00420/06	00047/07	00078/07	00079/07	00084/07	00087/07	00139/07
	00143/07	00147/07	00150/07				
ID2	00530/07	00518/02	00540/02	00133/04	00099/05	00625/05	00713/05
	00192/06	00204/06	00141/07	00158/07	00162/07		
ID3	00531/07	00522/02	00542/02	00445/03	00447/03	00595/03	00598/03
	00137/04	00367/04	00370/04	00461/04	00101/05	00350/05	00352/05
	00634/05	00717/05	00196/06	00208/06	00213/06	00006/07	00172/07
	00180/07						
ID4	00532/07	00549/02	00599/03	00149/04	00103/05	00175/07	00184/07
	00190/07						
ID5	00533/07	00552/02	00205/04	00105/05	00036/07	00168/07	00176/07
	00194/07						
ID6	00534/07	00118/02	00565/02	00004/03	00040/03	00048/03	00064/03
	00128/03	00129/03	00448/03	00458/03	00461/03	00633/03	00450/04
	00510/04	00107/05	00347/05	00627/05	00424/06	00038/07	00053/07
	00058/07						
ID7	00535/07	00485/01	00491/01	00123/02	00568/02	00082/03	00640/03
	00119/04	00109/05	00040/07	00052/07	00059/07		
IDCLR	00482/03	00444/03					
IDEC	00757/05	00727/05					
IDMBS	00588/07	00604/03	00643/03				
IDSCN	00415/06	00593/03	00419/06	00429/06	00430/06	00431/06	
IDX	00092/05	00257/01	00506/02	00062/03	00076/03	00443/03	00629/03
	00364/04	00457/04	00465/04	00096/05	00118/05	00126/05	00345/05
	00621/05	00186/06	00418/06				
IGNOR	00589/07	00249/02	00273/02	00292/02	00304/02	00323/02	00351/02

00364/02 00367/02 00371/02 00509/02

IMAIN 00590/07 00055/03 00073/03 00632/03 00619/05 00421/06

INCRA 00382/06 00387/06

INDSK 00137/06 00155/06 00172/06

INIDX 00081/05 00256/01 00505/02 00442/03 00628/03 00084/05 00185/06

INLST 00043/05 00253/01 00141/02 00159/02 00201/02 00266/02 00375/02
00398/02 00687/02 00288/04 00337/04 00392/04 00554/04 00005/05
00046/05 00363/05 00399/05 00723/05 00290/06

INNAM 00470/05 00206/04 00490/05

INREL 00530/05 00464/05 00488/05 00548/05

INSCN 00591/07 00522/04 00548/04

INSTR 00592/07 00569/04 00571/04 00579/04 00601/04

IP 00736/05 00728/05

JBINC 00711/07 00278/01 00280/01 00283/01

JBINS 00710/07 00274/01

JBN10 00288/01 00305/01

JBN20 00301/01 00296/01

JBN30 00306/01 00298/01

L001 00414/07 00094/04 00094/04 00105/04 00240/07 00343/07

L01 00415/07 00104/04 00112/04 00154/04 00164/04 00199/04 00260/04
00310/04 00422/04 00608/04 00538/04 00234/07 00303/07 00339/07

LAB10 00021/06 00018/06

LAB19 00129/06 00099/06 00104/06 00106/06

LAB20 00041/06 00036/06

LAB25 00057/06 00051/06

LAB30 00059/06 00048/06 00054/06

LAB40 00061/06 00045/06

LABDO 00011/06 00309/04 00542/04 00031/05 00066/06 00068/06 00400/07

LBERR 00287/02 00265/02

LBUF 00507/07 00564/01 00097/02 00477/07

LCNT 00593/07 00002/02 00100/02 00109/02 00258/03 00353/03 00375/03
00408/03 00243/04 00245/04 00462/05 00542/05

LD1 00101/04 00096/04

LD2 00111/04 00107/04

LD3 00091/04 00374/07

LD4 00160/04 00156/04

LD5 00163/04 00159/04

LD6 00205/04 00201/04

LD7 00265/04 00262/04

LD8 00543/04 00540/04

LDABS 00553/03 00536/03

LDBLR 00292/02 00282/02

LDING 00594/07 00592/03 00119/05

LDRIN 00560/01 00574/01 00072/02 00088/02

LENDR 00367/02 00286/02

LENTN 00297/02 00280/02

LEXTR 00298/02 00284/02

LIB1 00262/02 00384/02

LIBRY 00702/05 00590/03 00613/03 00657/03 00222/04 00235/04 00680/05
00733/05

LIBS 00244/02 00213/02 00245/02

LIBUT 00476/04 00471/04 00473/04

LINKC 00590/04 00337/07 00416/07

*LINNM 00279/02

LIRD 00260/02 00365/02

LISKP 00277/02 00275/02 00295/02 00309/02 00350/02

LISTU 00652/07 00145/01 00558/03 00181/05

LMAIN 00440/03 00200/02 00203/02 00219/02 00236/02

LO01 00178/01 00170/01

LOAD	00071/04	00607/03	00079/04				
LOAD1	00168/01	00609/06					
LOADL	00159/02	00183/02	00193/02	00197/02			
LOADN	00090/04	00373/04					
LOADR	00136/01	00005/01	00508/07	00510/07	00511/07	00512/07	00730/07
LOADS	00087/04	00652/03	00078/04	00326/04	00328/04		
LOADX	00104/04	00280/07	00351/07				
LODLB	00138/02	00519/01	00522/01	00556/01			
LODLL	00198/02	00161/02	00216/02				
LODLX	00184/02	00165/02	00169/02	00174/02			
LOVER	00542/06	00075/05					
LOWER	00595/07	00370/05	00374/05	00379/05	00382/05	00287/06	00294/06
	00304/06						
LPARN	00464/07	00460/06	00505/06				
LSCAN	00397/02	00321/02	00400/02	00417/02	00418/02	00240/06	
LST1	00518/07	00260/01	00146/02	00163/02	00402/02	00424/02	00691/02
	00016/03	00027/03	00340/04	00396/04	00428/04	00008/05	00060/05
	00412/05	00726/05	00049/07	00075/07	00088/07		
LST10	00400/05	00404/05	00415/05				
LST2	00519/07	00262/01	00148/02	00167/02	00406/02	00426/02	00695/02
	00029/03	00344/04	00400/04	00430/04	00012/05	00062/05	00742/05
	00050/07	00073/07	00094/07	00098/07			
LST20	00416/05	00401/05					
LST3	00520/07	00264/01	00151/02	00170/02	00180/02	00182/02	00186/02
	00188/02	00344/02	00378/02	00380/02	00409/02	00429/02	00698/02
	00031/03	00291/04	00293/04	00348/04	00406/04	00432/04	00445/04
	00557/04	00016/05	00064/05	00746/05	00242/06	00244/06	
LST4	00521/07	00266/01	00153/02	00176/02	00189/02	00196/02	00204/02
	00269/02	00329/02	00333/02	00340/02	00358/02	00710/02	00021/03
	00046/03	00067/03	00081/03	00357/04	00450/04	00483/04	00492/04
	00022/05	00066/05	00356/05	00402/05	00245/06	00247/06	
LST5	00522/07	00267/01	00154/02	00179/02	00185/02	00342/02	00356/02
	00033/03	00417/04	00420/04	00446/04	00449/04	00562/04	00025/05
	00068/05	00360/05	00751/05	00249/06	00293/06	00297/06	00302/06

LSTCR 00295/04 00290/04

LSTEX 00396/05 00217/02 00418/05 00419/05 00420/05

LSTX 00055/05 00254/01 00142/02 00160/02 00202/02 00267/02 00376/02
00399/02 00688/02 00289/04 00338/04 00393/04 00555/04 00006/05
00059/05 00074/05 00354/05 00400/05 00724/05 00291/06

LTERM 00022/04 00547/06

LTERW 00057/04 00053/04

LTERX 00043/04 00039/04

LTERZ 00051/04 00027/04 00032/04

LTOD 00101/02 00110/02

LWH1 00409/07 00556/02 00137/03 00166/03 00093/04 00263/04 00377/07

LWH2 00410/07 00626/02 00148/03 00282/07 00289/07 00347/07 00372/07
00383/07 00403/07

LWH3 00411/07 00100/04 00108/04 00115/04

LWH4 00412/07 00098/04 00111/04 00195/04 00516/04

LWR 00175/07 00197/07

M0760 00436/07 00254/02 00582/04 00585/04 00294/07 00361/07

M1100 00433/07 00560/03

*M17 00701/07

M1740 00437/07 00564/04 00576/04

M177 00704/07 00281/01 00119/02 00562/02 00005/03 00041/03 00049/03
00067/03 00153/03 00321/03 00449/03 00507/03 00515/03 00636/03
00228/04 00469/04 00511/04 00577/05 00601/05 00628/05 00714/05
00719/05 00778/05 00274/06 00426/06 00056/07 00145/07 00148/07

M1774 00706/07 00447/01 00150/02 00171/02 00181/02 00187/02 00379/02
00410/02 00413/02 00428/02 00488/02 00523/02 00683/02 00699/02
00446/03 00459/03 00138/04 00292/04 00349/04 00352/04 00484/04
00407/04 00433/04 00017/05 00258/05 00351/05 00635/05 00747/05
00092/06 00121/06 00139/06 00197/06 00209/06 00243/06 00476/06
00519/06

*M1777 00708/07

M200 00467/07 00632/05

M2000 00435/07 00595/04 00600/04 00363/07

M240 00569/01 00567/01

M300 00468/07 00138/01 00230/02
M37 00702/07 00301/02 00669/02 00378/04
M377 00705/07 00285/01 00238/04 00552/04 00558/04 00252/05 00581/05
00087/06 00096/06 00126/06 00144/06 00454/06 00474/06
M3777 00707/07 00243/01
M400 00469/07 00325/01 00330/01 00259/05 00010/07
M60 00466/07 00391/06
M600 00431/07 00144/01
M700 00432/07 00558/06
M7600 00470/07 00649/05 00660/05 00269/06
M77 00703/07 00520/01 00612/02 00559/03 00499/04 00553/06
MAXAD 00596/07 00064/06 00067/06 00174/06
MAXC 00421/07 00569/02 00618/02 00627/02 00396/07 00399/07 00401/07
MAXWK 00127/02 00113/02 00121/02
MESAB 00061/04 00545/06
MESLD 00056/02 00549/01
MESSE 00156/05 00151/05 00153/05
MESUX 00421/05 00409/05
MLIST 00510/07 00193/04 00194/04 00427/04 00429/04 00431/04 00435/04
00478/07 00479/07
MSECT 00726/07
MSIGN 00438/07 00547/02 00149/03 00358/04 00493/04 00565/04 00578/04
00596/04 00033/05
MNO 00689/07
N1 00688/07 00205/02 00270/02 00354/03 00502/03 00613/05
N10 00679/07 00566/03
N128 00472/07 00532/01 00001/02 00115/02 00355/03 00378/03 00493/03
00540/03 00575/03 00170/04 00177/04 00313/04 00476/04 00212/05
00310/05 00541/05 00685/05 00695/05 00781/05
N13 00567/05 00557/05

N19 00424/07 00080/06
N2 00687/07 00224/01 00231/05 00284/05 00606/05 00225/06 00364/06
N3 00686/07
N4 00685/07 00227/06
N5 00684/07 00521/04
N6 00683/07 00197/01 00344/03 00516/03
N7 00682/07 00116/05
N72 00423/07 00347/01 00029/07
N8 00681/07 00398/06 00070/07
N9 00680/07 00560/02
NAENT 00460/07 00572/03
NAM 00475/02 00462/01 00069/02 00276/02 00576/02
NAM12 00580/02 00482/02 00513/02 00537/02
NAM34 00581/02 00485/02 00517/02 00539/02
NAM5 00582/02 00489/02 00521/02 00541/02
NAMR 00059/02 00024/02
NB60 00629/06 00066/07
NBPF 00598/07 00196/01 00386/05
NBPL 00584/02 00495/02 00574/02
NCL 00585/02 00498/02 00548/02
NCPL 00279/07 00246/07 00264/07 00266/07
NDA 00592/05 00265/03 00364/03 00533/05 00598/05 00611/05
NDA2 00609/05 00603/05
NEGPM 00536/06 00361/01 00385/01 00443/06 00482/06 00487/06 00033/07
00117/07
NEW1 00367/03 00372/03
NEW2 00371/03 00360/03
NEWS 00374/03 00326/03
NEXT 00688/02 00693/02 00597/02 00702/02

NEXTD 00370/06 00378/06
NFRST 00223/05 00217/05
NLENT 00442/04 00424/04
NMERR 00039/02 00026/02 00060/02 00214/04 00259/04
NODBG 00157/02 00140/02
NOENT 00026/03 00689/02
NOLOD 00065/04 00012/04
NPL 00583/02 00492/02 00553/02
NSYMB 00312/02 00349/02
NTFND 00537/02 00507/02
NTMN 00070/03 00066/03
NTYPE 00586/02 00501/02 00559/02
NXFLG 00599/07 00299/02 00307/02 00326/02 00335/02 00345/02 00666/02
00706/02 00037/03 00084/03 00376/04 00411/04
NXSYM 00383/04 00496/04
NXTID 00506/02 00511/02 00516/02 00520/02 00526/02
O1777 00434/07 00178/02 00592/04 00599/04 00250/07
OMITE 00373/02 00369/02
OPRND 00601/07 00416/04 00436/04 00588/04 00378/05 00431/05 00320/07
00335/07
OPWRS 00485/07 00357/06
OUTID 00764/05 00538/03 00544/03 00023/04 00624/05 00626/05 00637/05
00639/05 00664/05 00666/05 00668/05 00670/05 00672/05 00679/05
00683/05 00692/05 00698/05 00768/05 00783/05
OUTR 00071/02 00075/02 00080/02
PAGNO 00603/07 00583/04 00586/04 00308/07
PBE 00620/06 00606/06
PBREL 00604/07 00183/01 00617/03 00676/03 00680/03 00076/04 00653/05
PBUF 00512/07 00365/01 00440/06 00441/06 00442/06 00512/06 00515/06
00518/06 00482/07

PCNTT 00605/07 00307/01 00425/01 00499/06 00522/06

PCONT 00535/06 00501/06 00513/06 00514/06 00516/06 00517/06 00520/06
00521/06

PENTZ 00285/04 00267/04 00276/04

PGCNT 00606/07 00591/03 00610/03 00008/04 00030/04

PGMIN 00607/07 00139/01 00318/01 00517/01 00563/01 00231/02 00552/06

PIDNT 00526/07 00200/01 00541/01 00045/02 00572/02 00072/05 00094/05
00327/07

PLAD 00483/07 00430/01 00587/03 00004/04 00172/04 00174/04 00315/04
00478/04 00328/05 00333/05 00214/06 00500/06 00046/07 00082/07
00138/07

PLCNT 00608/07 00171/04 00178/04 00184/04 00190/04 00314/04 00322/04
00477/04 00486/04 00336/05 00341/05

PLEXT 00479/04 00487/04

PLFLG 00609/07 00530/01 00022/02 00065/02 00083/02 00089/04 00506/04
00618/05

PLGTH 00610/07 00218/04 00295/04 00052/06 00267/07 00356/07

PLIST 00508/07 00043/07 00044/07 00045/07 00121/07 00122/07 00123/07
00483/07

PLMAN 00611/07 00627/03 00176/04 00179/04 00330/05

PLSCN 00316/04 00323/04

PLST 00516/07 00192/01 00269/01 00543/01 00047/02 00156/02 00431/02
00036/03 00057/05 00113/05 00312/07 00393/07

PMBUF 00537/06 00354/01 00449/06 00457/06

PPREL 00612/07 00557/03 00583/03 00602/03 00074/04 00640/05

PRBLN 00098/07 00062/07

PRCMA 00093/07 00064/07

PRENT 00613/07 00286/04 00030/05 00037/05 00678/05 00737/05

PRENX 00310/02 00306/02

•PRHED 00567/03

PRN 00104/07 00096/07 00100/07

PROG5 00508/06 00533/06

PROG9 00532/06 00526/06

PROGL 00497/06 00379/01 00504/06 00507/06 00511/06 00528/06 00531/06
00534/06

PS1 00283/07 00277/07

PS2 00282/07 00242/07

PS3 00291/07 00238/07

PTYPE 00614/07 00441/03 00477/03 00482/03 00582/03 00624/03 00002/04
00167/04 00324/04 00329/05 00641/05 00427/06

RANAD 00616/07 00360/06 00382/06 00388/06 00390/06

RBTAD 00499/07 00532/04

RCERR 00036/02 00423/01 00474/01 00063/03 00077/03 00319/03 00331/03
00335/03 00346/03 00339/04 00365/04 00007/05 00024/05 00346/05
00622/05

RCN02 00328/01 00323/01

RCN04 00336/01 00327/01

RCN05 00342/01 00335/01

RCN10 00358/01 00388/01

RCN30 00379/01 00372/01

RCN50 00511/01 00320/01 00357/01 00361/01 00384/01

RCN60 00529/01 00061/02 00229/02 00232/02

RCNTL 00321/01 00276/01 00279/01 00349/01 00387/01 00239/02

RCNTP 00316/01

REKEY 00618/07 00520/04 00524/04 00526/04

RELAD 00617/07 00556/03 00221/04 00270/04 00299/04 00302/04 00014/06
00061/06

REMD0 00161/06 00609/03 00654/03 00037/06 00138/06 00143/06 00150/06
00166/06

RES05 00252/06 00184/06

RES15 00186/06 00203/06 00207/06 00212/06 00223/06

RES20 00200/06 00191/06 00195/06

RES30 00213/06 00199/06 00220/06

RES40 00221/06 00218/06
RES60 00224/06 00187/06
RES60 00231/06 00251/06
RES70 00250/06 00241/06
RESET 00177/06 00611/03 00655/03 00265/06
PRIC 00619/07
RLGTH 00620/07 00568/02 00323/03
RPARN 00465/07 00462/06 00524/06
RQ 00301/05 00289/05 00293/05
RRL9 00267/06 00024/06 00255/06 00276/06
RSS 00213/04 00101/04
RTRN 00284/03 00192/03 00282/03
SAVNM 00676/02 00088/03
SAVSS 00622/07 00171/01 00054/04
SBFWA 00724/07 00160/01 00171/07
SBLWA 00725/07 00162/01 00179/07
SCANX 00358/02 00325/02
SCHK 00086/07 00081/07
SCNT 00382/03 00327/03 00379/03 00406/03 00427/03
SDSK 00053/02 00546/01 00048/02
SECT 00586/05 00293/01 00294/01 00468/03 00469/03 00236/06 00298/05
00518/05 00578/05
SECTR 00714/07 00303/01 00253/05
SELST 00422/02 00338/02 00354/02 00432/02
SET 00551/06 00050/02 00018/04 00555/06 00557/06 00564/06
SETCR 00070/06 00380/01 00495/03 00024/04 00769/05 00073/06 00109/06
00119/06 00129/06 00149/06 00162/06 00334/06
SETL 00307/07 00367/07
SETMP 00192/04 00186/04

SETOP 00427/05 00421/04 00590/04 00434/05
SETUP 00191/03 00411/01 00246/02 00202/03
SEXT 00489/04 00481/04
SIDA 00020/03 00001/03 00008/03 00039/03
SIDNT 00055/02 00542/01 00044/02
SKIP 00425/03 00456/01 00464/01 00493/01 00277/02 00373/02 00380/03
00429/03 00435/03
SLASH 00490/06 00464/06
SLSHE 00444/07 00356/01 00360/01 00383/01 00486/06
SLST 00054/02 00544/01 00046/02
SMFWA 00723/07 00156/01 00174/07
SMLWA 00709/07 00158/01 00198/01 00102/07
SPACE 00162/05 00570/03 00574/03 00606/03 00651/03 00007/04 00166/05
SSFLG 00729/07 00168/01 00051/04 00671/07
STAT 00568/05 00559/05
STB 00004/07 00594/06 00605/06 00069/07 00072/07 00065/07 00188/07
00191/07 00192/07
STB1 00603/06 00624/06
STB2 00008/07 00031/07
STB3 00024/07 00015/07
STB4 00042/07 00105/07
STB5 00052/07 00091/07 00102/07
STBND 00592/06 00155/01 00617/06
SUBH5 00234/04 00230/04
SUBHD 00195/04 00158/04 00161/04
SYDBL 00497/01 00469/01
SYENT 00500/01 00467/01
SYEXT 00501/01 00471/01
SYM 00143/05 00136/05

SYM10	00418/01	00494/01						
*SYM12	00623/07							
SYM20	00432/01	00455/01						
SYM30	00451/01	00437/01	00443/01	00450/01				
*SYM34	00624/07							
SYM40	00456/01	00460/01						
*SYM5	00625/07							
SYM50	00462/01	00427/01	00449/01					
SYM70	00464/01	00498/01	00503/01					
SYM75	00493/01	00459/01						
SYMBL	00408/01	00312/01	00381/01	00410/01				
SYNTS	00719/07	00481/07						
SYOUT	00135/05	00377/01	00550/01	00532/02	00017/03	00476/03	00013/04	
	00017/04	00145/05	00154/05	00410/05	00413/05	00546/06		
SYSSC	00717/07	00239/01	00036/04					
SYSTY	00713/07	00324/01	00329/01	00333/01	00616/06	00623/06	00009/07	
	00013/07							
TABS	00627/07	00012/06	00043/06	00059/06				
TADR	00628/07	00013/06	00023/06					
TBREL	00629/07	00182/01	00616/03	00621/03	00658/03	00662/03	00665/03	
	00077/04	00099/04	00109/04	00116/04	00371/05	00441/05	00446/05	
	00447/05	00658/05	00671/05					
TBUF	00511/07	00313/02	00316/02	00319/02	00343/02	00401/02	00405/02	
	00411/02	00412/02	00414/02	00423/02	00425/02	00427/02	00677/02	
	00680/02	00684/02	00690/02	00694/02	00700/02	00026/03	00028/03	
	00030/03	00362/04	00371/04	00384/04	00387/04	00390/04	00395/04	
	00399/04	00403/04	00444/04	00456/04	00460/04	00462/04	00463/04	
	00467/04	00499/04	00645/05	00652/05	00656/05	00663/05	00665/05	
	00669/05	00178/06	00232/06	00235/06	00238/06			
TCNT	00630/07	00361/03	00365/03	00431/03	00433/03	00027/06	00032/06	
	00228/06	00250/06	00259/06	00262/06	00328/06	00340/06	00342/06	
	00365/06	00377/06						
TDIR	00631/07	00311/01	00317/01	00369/01	00408/01	00195/03	00122/04	
	00596/05							
TEMP	00632/07	00219/01	00326/01	00331/01	00345/01	00071/03	00079/03	

	00262/03	00267/03	00542/03	00545/03	00561/03	00565/03	00336/04
	00356/04	00028/05	00032/05	00034/05	00176/05	00183/05	00694/05
	00699/05	00716/05	00720/05	00721/05	00743/05	00230/06	00231/06
	00233/06	00234/06	00236/06	00237/06	00239/06	00559/06	00563/06
	00011/07	00027/07					
TEMP1	00407/07	00006/02	00020/02	00613/02	00624/02	00142/03	00170/03
	00261/07	00274/07	00284/07	00291/07	00369/07		
TEMP2	00408/07	00126/04	00157/04	00537/04	00541/04	00255/07	00268/07
TERMN	00633/07	00359/01	00382/01	00303/02	00348/02	00502/06	00509/06
	00523/06	00530/06					
TESTR	00577/01	00571/01					
TIDNT	00525/07	00571/02	00061/03	00074/03	00646/03	00669/03	00361/04
	00454/04	00464/04	00083/05	00093/05	00110/05	00111/05	00117/05
	00344/05	00620/05	00417/06	00422/06			
TLOG	00569/05	00560/05					
TLST	00515/07	00268/01	00155/02	00430/02	00034/03	00045/05	00056/05
	00070/05						
TNEXT	00634/07	00034/06	00038/06				
TPREL	00501/07	00619/03	00667/03	00075/04	00097/04	00114/04	00220/04
	00234/04	00272/04	00297/04	00300/04	00308/04	00414/04	00536/04
	00572/04	00580/04	00036/05	00647/05	00667/05	00682/05	00254/06
	00248/07	00283/07	00286/07	00358/07	00382/07	00389/07	
TR	00587/05	00218/01	00222/01	00227/01	00233/01	00287/01	00292/01
	00295/01	00299/01	00301/01	00235/05	00297/05	00517/05	00582/05
TREAD	00548/01	00575/01					
TRID3	00337/05	00349/05	00355/05				
TSONE	00721/07	00505/03	00776/05	00114/06	00153/06	00338/06	
TTBUF	00266/06	00179/06	00262/06				
TTYNO	00635/07	00138/05	00144/05				
TWORD	00636/07	00025/06	00042/06				
UBFWA	00646/07	00161/01	00181/01	00195/01	00369/05	00596/06	
UBLNK	00442/07	00748/05	00367/06				
UBLWA	00647/07	00163/01	00186/01	00450/05	00597/06		
UDNTS	00718/07	00247/01	00046/04	00480/07			
UMFWA	00648/07	00157/01	00553/03	00279/04	00607/06		

UMLWA 00649/07 00159/01 00016/06 00608/06
UPDA1 00110/06 00120/06
UPDAT 00107/06 00101/06
UPNAM 00136/03 00374/02 00130/03 00139/03 00173/03
UPPER 00638/07 00373/05 00375/05 00289/06 00298/06 00306/06
UPPR 00179/07 00170/07
UPPR1 00184/07 00196/07
USCHK 00462/03 00451/03 00455/03
PUTCNT 00639/07
PUTRSC 00640/07
W.PRG 00506/01 00419/01 00477/01 00404/03
WAIT 00553/05 00223/05 00537/05 00563/05 00565/05
WCMND 00240/05 00232/05
WDCNT 00642/07 00203/01 00208/01 00429/01 00454/01 00012/02 00156/03
00162/03 00332/03 00311/05 00315/05 00310/06 00316/06
WQWC 00493/05 00476/05
XPREL 00644/07 00584/03 00601/03 00448/06 00466/06 00481/06
XSSFL 00671/07 00173/01 00055/04
XXXX1 00411/04 00394/04
ZBUF 00653/07 00132/04 00136/04 00139/04 00609/04 00613/04 00001/05